

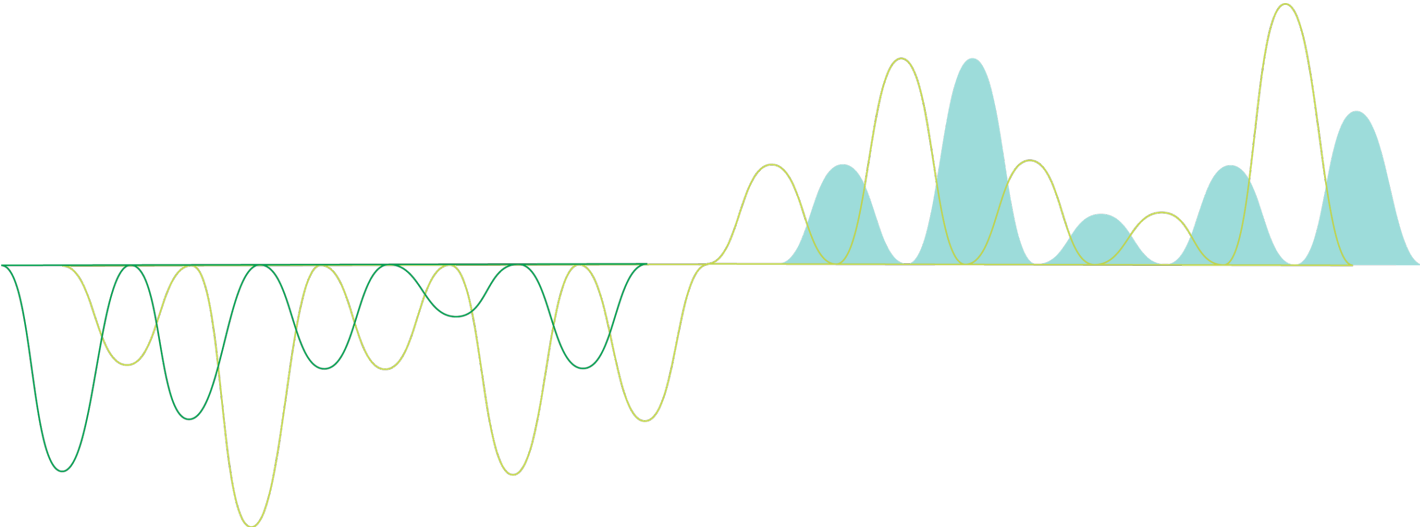
Qlik Replicate Setup and User Guide

Qlik Replicate™

May 2023

Last Updated: May 16, 2024

Copyright © 1993-2024 QlikTech International AB. All rights reserved.



1 What's new?	26
1.1 New endpoints and endpoint enhancements in Replicate May 2023 service release 01	26
New Confluent Cloud target endpoint	26
New authentication option for Kafka target	26
1.2 Endpoints and endpoint enhancements first introduced in Replicate May 2023 initial release	26
New Google Cloud Pub/Sub target endpoint	26
New IBM DB2 for z/OS target endpoint	27
New Google Cloud AlloyDB for PostgreSQL endpoints	27
SAP ODP source endpoint enhancements	27
IBM DB2 for z/OS source endpoint - data server client support	27
AWS Aurora Cloud for PostgreSQL source endpoint: non-superuser support	28
Log Stream tasks	28
Endpoint proxy server enhancements	28
Microsoft Azure Data Lake Storage (ADLS) Gen2	28
Amazon Redshift target endpoint - AWS PrivateLink support	28
Support for Unity Catalog with Databricks endpoints	28
Data type enhancements	29
Extended Parallel Load support	29
Default quote character in file-based endpoints	29
Renamed IBM Netezza target endpoint	29
Microsoft Azure SQL (MS-CDC) endpoint - Geo Replica support	29
Oracle HSM (Hardware Security Module) support	30
1.3 What's new on the server side?	30
Default proxy server	30
Support special characters in column names used in expressions	30
2 Introduction	31
2.1 What is Qlik Replicate?	31
System architecture	32
2.2 Terminology	32
Change Data Capture (CDC)	32
Full load	33
Apply latency	33
Source latency	33
Target latency	33
Overall latency	33
Source endpoint	33
Target endpoint	34
Net Changes table	34
2.3 Replication Explained	34
2.4 Full Load and CDC processes	35
2.5 Replication topologies	36
One to one	36
Logical independence	36
Hub and spoke	36
2.6 Limitations and considerations	36
3 Installing Qlik Replicate	39
3.1 Recommended hardware configuration	39

3.2	Setting up Qlik Replicate on Windows	40
	Windows permissions	41
	Windows software requirements	44
	Installing or upgrading Qlik Replicate on Windows	44
	Starting and stopping the Qlik Replicate Server service on Windows	46
	Silently installing Qlik Replicate	46
	Silently upgrading Qlik Replicate	47
	Silently uninstalling Qlik Replicate	48
	Changing the data directory location on Windows	49
3.3	Setting up Qlik Replicate on Linux	49
	Replicate instances and services on Linux	50
	Linux installation prerequisites and procedures	51
	Replicate server procedures	53
	Upgrading Qlik Replicate	54
	Uninstalling Qlik Replicate	56
	Working with additional Replicate instances	57
4	Security considerations	60
4.1	Securing access to the Qlik Replicate Web UI	60
4.2	Setting up HSTS on Replicate	61
	Enabling HSTS on the Replicate UI Server	61
	Disabling HSTS on the Replicate UI Server	61
	Configuring HSTS on the Replicate C UI Server	62
4.3	Setting Up Qlik Replicate console HTTPS support	63
	Checking if an SSL certificate is installed	63
	Using the self-signed certificate	64
4.4	Setting Up Qlik Replicate Server HTTPS support	64
	Replacing the self-signed SSL certificates on Linux	64
	Replacing the self-signed certificate on Windows	65
4.5	FIPS compliance	66
	FIPS-compliant endpoints	67
4.6	TLS Support	68
4.7	Changing the server password	69
4.8	Protecting Replicate passwords	71
	The master key file	71
	Changing and protecting the master key	72
4.9	Client password considerations	74
4.10	Encrypting the user permissions file	74
4.11	Securing connections to endpoints	75
4.12	Log file encryption	75
4.13	Application security	77
4.14	Using external credentials	77
	Limitations and considerations	77
	Supported endpoints	77
	Changing the default addon name	78
	Step 1: Edit the MyPasswordProvider.c file	78
	Step 2: Compile the code	78

Step 3: Edit the addons_def.json.sample file	79
Step 4: Configure the endpoints	79
4.15 Controlling execution of user-defined commands	79
Executing operating system commands as a different user	80
5 Working with Qlik Replicate endpoints	81
5.1 Using ARC CDC agents as endpoints	81
5.2 Replicate data types	81
5.3 Supported DDL statements	83
How Replicate handles DDL changes	83
Limitations and considerations	84
5.4 Configuring Replicate to automatically replace the user-entered password	85
Defining multiple endpoints to use the same automatically changed password	85
6 Using the Qlik Replicate Console	87
6.1 Opening the Qlik Replicate console	87
Accessing Qlik Replicate from a remote computer	87
Qlik Replicate UI server configurations	89
Setting up Qlik Replicate UI Server to work with a remote Qlik Replicate Server	90
6.2 Tasks view	91
Viewing specific tasks	93
6.3 Server view	96
6.4 List actions	96
7 Getting started: A Qlik Replicate tutorial	97
7.1 What you need	97
7.2 Open the Qlik Replicate console	97
7.3 Add an Oracle endpoint as a source	98
7.4 Add a Microsoft SQL Server database as a target	99
7.5 Define a replication task	102
Add a replication task to the Qlik Replicate console	102
Add the source and target endpoints to the task	104
Select tables for the Replication task	106
7.6 Run and monitor the replication task	107
7.7 View the replicated tables in Microsoft SQL Server	108
8 Defining and managing tasks	109
8.1 Tasks overview	109
Using multiple tasks	110
8.2 Adding tasks	110
Bidirectional replication	111
Editing and viewing a task description	114
8.3 Defining and managing endpoints	114
Defining an endpoint	115
Editing endpoint configuration information	115
Viewing endpoint configuration information	116
Testing an endpoint connection	116
Duplicating endpoints	116
Searching for endpoints	117
Deleting endpoints	117

8.4 Adding a source and target endpoint to a task	117
8.5 Selecting tables and/or views for replication	118
Searching for tables/views to use in a replication task	120
Selecting specific tables/views for replication	121
Removing specific tables/views from a replication task	122
Creating table/view selection patterns	122
Setting load order	124
8.6 Editing a replication task	125
8.7 Deleting a replication task	125
8.8 Searching for tasks	126
8.9 Migrating tasks	126
Exporting tasks	126
Importing tasks	127
Editing an exported (json) file	129
9 Adding and managing source endpoints	131
9.1 Using AWS Aurora Cloud for PostgreSQL as a source	132
Prerequisites	133
Required permissions	133
Using an account without the "superuser" role	134
Setting up AWS Aurora Cloud for PostgreSQL for CDC (Change data capture)	135
Limitations and considerations	136
Supported source data types	137
Homogeneous Replication	140
Setting general connection properties	142
Setting advanced connection properties	144
Removing Replicate artifacts from the AWS Aurora Cloud for PostgreSQL database	145
9.2 Using Amazon RDS for SQL server as a source	146
Prerequisites	147
Limitations and considerations	148
Required permissions	149
Supported compression methods	149
Amazon RDS for SQL server source data types	150
Non-supported data types	152
Homogeneous Replication	153
Preparing the Amazon RDS for SQL server database for Replication	154
Setting general connection properties	156
Setting advanced connection properties	158
Artifacts created by Replicate in the database	158
9.3 Using Amazon RDS for MySQL as a source	160
Prerequisites	161
Limitations and considerations	162
Permissions	163
Data types	163
Homogeneous Replication	166
Setting general connection properties	167
Selecting a schema	169
Setting advanced connection properties	169

9.4 Using Amazon RDS for PostgreSQL as a source	170
Prerequisites	171
Required permissions	172
Using an account without the "superuser" role	172
Setting up Amazon RDS for PostgreSQL for CDC (Change data capture)	174
Limitations and considerations	174
Supported data types	175
Homogeneous Replication	179
Setting general connection properties	181
Setting advanced connection properties	182
Removing Replicate artifacts from the Amazon RDS for PostgreSQL database	183
9.5 Using ARC CDC solutions in Qlik Replicate	185
Prerequisites for using ARC CDC solutions	186
ARC CDC solution security considerations	187
Limitations	189
ARC source data type mapping	189
Working with ARC CDC solutions	190
Setting advanced connection properties	193
9.6 Using a file as a source	195
General overview	196
File source overview	196
Prerequisites	199
Limitations	199
Setting general connection properties	199
Setting advanced options	207
9.7 Using Google Cloud AlloyDB for PostgreSQL as a source	211
Prerequisites	212
Limitations and considerations	213
Supported data types	214
Homogeneous replication	218
Setting general connection properties	220
Setting advanced connection properties	221
Removing Replicate artifacts from the database	223
9.8 Using Google Cloud SQL for MySQL as a source	224
Prerequisites	225
Limitations and considerations	226
Permissions	227
Data types	227
Homogeneous Replication	230
Setting general connection properties	231
Selecting a schema	233
Setting advanced connection properties	233
9.9 Using Google Cloud SQL for PostgreSQL as a source	234
Prerequisites	235
Required permissions	236
Limitations and considerations	236
Supported data types	237
Homogeneous Replication	241

Setting general connection properties	243
Setting advanced connection properties	244
Removing Replicate artifacts from the Google Cloud SQL for PostgreSQL database	246
9.10 Using Hadoop as a source	247
Prerequisites	248
Limitations and considerations	248
Required permissions	248
Hadoop Endpoint source data types	248
Setting general connection properties	249
Setting advanced connection properties	253
9.11 Using IBM DB2 for iSeries as a source	256
Prerequisites	257
Required permissions	259
Limitations and considerations	259
Supported data types	260
Setting general connection properties	261
Setting advanced connection properties	262
9.12 Using IBM DB2 for LUW as a source	267
Prerequisites	268
Replicating 4-byte UTF8 emojis	269
Limitations and considerations	269
Supported data types	270
Setting general connection properties	272
Setting advanced connection properties	273
9.13 Using IBM DB2 for z/OS as a source	276
Prerequisites	277
Limitations and considerations	280
Handling actions resulting in subtype 83	281
Supported data types	282
Setting general connection properties	284
Setting advanced connection properties	285
Sample XMIT files “Receive” job	289
Installing and configuring R4Z	290
9.14 Using IBM Informix as a source	306
Prerequisites	307
Limitations and considerations	307
Required permissions	308
Supported data types	308
Setting general connection properties	309
Setting advanced connection properties	310
9.15 Using Microsoft Azure SQL Managed Instance as a source	312
Prerequisites	313
Limitations and considerations	316
Required permissions	317
Supported compression methods	317
Microsoft Azure SQL Managed Instance source data types	318
Non-supported data types	320
Homogeneous replication	321

Setting general connection properties	322
Setting advanced connection properties	323
Artifacts created by Replicate in the database	324
9.16 Using Microsoft Azure SQL (MS-CDC) as a source	325
Prerequisites	326
Limitations and considerations	327
Required permissions	329
Supported compression methods	329
Supported data types	330
Homogeneous replication	333
Setting general connection properties	334
Setting advanced connection properties	335
Removing Replicate artifacts from the database	336
9.17 Using Microsoft SQL Server (MS-CDC) as a source	337
Prerequisites	338
Limitations and considerations	339
Required permissions	341
Supported data types	342
Setting general connection properties	345
Setting advanced connection properties	346
Working with Windows Authentication	347
Removing Replicate artifacts from the database	347
9.18 Using Microsoft SQL Server as a source	348
Supported editions	349
Prerequisites	349
Limitations and considerations	350
Working with Microsoft SQL Server AlwaysOn Availability Groups	352
Required permissions	354
Supported compression methods	354
Microsoft SQL Server source data types	355
Homogeneous replication	358
Preparing the Microsoft SQL Server database for CDC	360
Setting general connection properties	362
Setting advanced connection properties	364
Working with Windows authentication	367
Artifacts created by Replicate in the database	367
9.19 Using Microsoft Azure Database for MySQL as a source	368
Prerequisites	369
Limitations and considerations	369
Permissions	370
MySQL database source data types	371
Homogeneous Replication	374
Setting general connection properties	375
Selecting a schema	376
Setting advanced connection properties	377
9.20 Using a MongoDB database as a source	378
Overview	379
Prerequisites	379

Limitations and considerations	380
Setting general connection properties	380
Setting advanced connection properties	383
9.21 Using a MySQL-based database as a source	385
Prerequisites	386
Limitations and considerations	388
Required permissions	389
MySQL database source data types	390
Homogeneous replication	393
Setting general connection properties	394
Selecting a schema	395
Setting advanced connection properties	395
9.22 Using ODBC to connect to a source	397
Prerequisites	398
Qlik Replicate Server for Linux	398
Limitations and considerations	398
ODBC source data types	399
Setting general connection properties	402
Setting advanced connection properties	404
9.23 Using ODBC with CDC as a source	405
Prerequisites	406
Limitations and Considerations	406
ODBC with CDC source data types	407
Setting general connection properties	410
Setting Change Processing parameters	412
9.24 Using Oracle as a source	416
Supported Oracle database editions	417
Prerequisites	417
Required permissions	420
Limitations and considerations	423
Supported encryption methods	426
Supported compression methods	427
Supported standby configurations	428
Redo log files - access method guidelines	428
Handling shrink space operations	431
Replicating nested tables	431
Oracle source data types	432
Unsupported data types	435
Homogeneous Replication	436
Working with Oracle on Oracle Cloud	437
Working with Amazon RDS for Oracle	437
Setting general connection properties	440
Setting advanced connection properties	442
Finding the wallet entries used for TDE Encryption	450
9.25 Using PostgreSQL as a source	452
Prerequisites	453
Required permissions	454
Using an account without the "superuser" role	454

Limitations and considerations	456
Supported data types	457
Homogeneous replication	461
Setting general connection properties	463
Setting advanced connection properties	464
Capturing changes from Microsoft Azure for PostgreSQL	466
Removing Replicate artifacts from the database	466
9.26 Using Salesforce (Streaming CDC) as a source	467
Prerequisites	468
Calculating the Salesforce quota	468
Limitations and considerations	470
Supported data types	471
Setting general connection properties	472
Setting advanced connection properties	473
9.27 Using Salesforce (Incremental Load) as a source	475
Prerequisites	476
Calculating the Salesforce quota	476
Limitations and considerations	477
Supported data types	478
Setting general connection properties	479
Setting advanced connection properties	480
9.28 Using SAP Application as a source	483
Prerequisites	484
Limitations and considerations	491
SAP Application source data types	491
Setting general connection properties	493
Setting advanced properties	496
9.29 Using SAP Application (DB) as a source	498
Prerequisites	499
Limitations and considerations	506
SAP Application (DB) source data types	507
Setting general connection properties	508
Setting advanced properties	509
9.30 Using SAP Extractor as a source	511
Overview	512
Prerequisites	512
Administration User	515
RFC Communications User	516
Deleting the remaining objects	520
Limitations and considerations	530
Supported data types	531
Setting general connection properties	532
Setting advanced properties	535
9.31 Using SAP HANA as a source	537
Prerequisites	538
Limitations and considerations	540
Required permissions	541
Supported data types	542

Setting general connection properties	543
Setting advanced properties	544
Removing Replicate artifacts from the database	548
9.32 Using SAP ODP as a source	549
Prerequisites	550
Limitations and considerations	552
Required permissions	552
Supported data types	555
Setting general connection properties	556
Setting advanced properties	559
9.33 Using SAP Sybase ASE as a source	562
Prerequisites	563
Limitations and considerations	563
Required permissions	564
SAP Sybase ASE database source data types	564
Setting general connection properties	566
Setting advanced connection properties	567
Handling the secondary truncation point	569
9.34 Using Teradata Database as a source	570
Prerequisites	571
Limitations and considerations	572
Required permissions	572
Teradata source data types	572
Setting general connection properties	573
Setting Change Processing parameters	574
10 Adding and managing target endpoints	579
10.1 Using Actian Vector as a target	581
Prerequisites	582
Limitations	583
Permissions	583
Actian Vector data types	583
Setting general connection properties	585
Setting advanced connection properties	586
10.2 Using Amazon MSK as a target	588
Transaction processing by the consumer	589
Prerequisites	590
Limitations	590
Amazon MSK target data types	591
Setting general connection properties	592
Setting advanced connection properties	597
Data messages	598
10.3 Using Amazon EMR as a target	601
Prerequisites	602
Limitations	602
Change data partitioning on Amazon EMR	603
Amazon EMR endpoint target data types	603
Setting general connection properties	604

Setting advanced connection properties	605
10.4 Using Amazon Kinesis data streams as a target	609
Prerequisites	610
Transaction processing by the consumer	611
Limitations	612
Supported data types	613
Setting general connection properties	615
Setting advanced connection properties	618
The Replicate envelope	619
Metadata and data messages	621
10.5 Using Amazon Redshift as a target	625
Introducing the Amazon Redshift target endpoint for Qlik Replicate	626
Limitations	626
Prerequisites	626
Supported data types	628
Setting general connection parameters	630
Setting advanced connection properties	632
10.6 Using Amazon S3 as a target	634
Prerequisites	635
Amazon S3 target overview	636
Limitations and considerations	637
Change data partitioning	637
Amazon S3 target data types	638
Setting general connection properties	640
Setting advanced connection properties	644
Generating reference files	647
Content-type and content-encoding properties	647
10.7 Using Cloudera Data Platform (CDP) Private Cloud as a target	649
Prerequisites	650
Limitations and considerations	650
Change data partitioning	651
Security requirements	651
Supported data types	651
Setting general connection properties	652
Setting advanced connection properties	655
Using Kerberos authentication on Windows	657
10.8 Using Confluent Cloud as a target	660
Overview	661
Prerequisites	662
Limitations and considerations	662
Supported data types	663
Setting general connection properties	664
Setting advanced connection properties	667
Data messages	668
10.9 Using Databricks Lakehouse (Delta) as a target	671
Prerequisites	672
Limitations and considerations	674
Supported data types	675

Setting general connection properties	676
Setting advanced connection properties	679
10.10 Using Databricks (Cloud Storage) as a target	682
Prerequisites	683
Limitations and considerations	685
Change data partitioning	686
Supported data types	686
Setting general connection properties	688
Setting advanced connection properties	694
10.11 Using a file as a target	697
File target overview	698
Limitations and considerations	699
Change data partitioning	700
File target data types	700
Setting general properties	701
Setting advanced connection properties	704
Generating reference files	708
10.12 Using Google Cloud AlloyDB for PostgreSQL as a target	709
Prerequisites	710
Limitations	711
Security requirements	711
Supported PostgreSQL data types	711
Setting general connection properties	712
Setting advanced connection properties	714
10.13 Using Google Cloud BigQuery as a target	715
Prerequisites	716
Limitations and considerations	717
Supported data types	718
Setting general connection properties	719
Setting advanced connection properties	720
10.14 Using Google Cloud Pub/Sub as a target	722
Overview	723
Prerequisites	724
Limitations	724
Supported data types	725
Setting general connection properties	726
Setting advanced connection properties	731
Data messages	732
10.15 Using Google Cloud SQL for MySQL as a Target	735
Prerequisites	736
Limitations and considerations	737
Security requirements	737
Supported Data Types	737
Setting general connection properties	740
Setting advanced connection properties	741
10.16 Using Google Cloud SQL for PostgreSQL as a target	743
Prerequisites	744

Limitations	745
Security requirements	745
Google Cloud SQL for PostgreSQL database target data types	745
Setting general connection properties	746
Setting advanced connection properties	748
10.17 Using Google Cloud SQL for SQL Server as a target	750
Supported editions	751
Prerequisites	751
Limitation and considerations	752
Permissions	753
Supported data types	753
Setting general connection properties	754
Setting advanced connection properties	755
10.18 Using Google Cloud Storage as a target	757
Prerequisites	758
Google Cloud storage target overview	758
Limitations and considerations	758
Change data partitioning	759
Data types	759
Setting general connection properties	761
Setting advanced connection properties	765
Generating reference files	767
10.19 Using Google Dataproc as a target	769
Prerequisites	770
Limitations	770
Change data partitioning on Google Dataproc	771
Google Dataproc endpoint target data types	771
Setting general connection properties	772
Setting advanced connection properties	773
10.20 Using Hadoop as a target	776
Prerequisites	777
Limitations	777
Change Data Partitioning on Hadoop	778
Security requirements	779
Hadoop endpoint target data types	779
Setting general connection properties	780
Setting advanced connection properties	785
Using Kerberos authentication	792
10.21 Using Hortonworks Data Platform (HDP) as a target	796
Prerequisites	797
Limitations	797
Change Data Partitioning	798
Security requirements	798
HDP Endpoint target data types	798
Setting general connection properties	799
Setting advanced connection properties	806
Using Kerberos authentication	809
10.22 Using IBM DB2 for z/OS as a target	813

Prerequisites	814
Limitations and considerations	816
Permissions	817
Supported target data types	817
Setting general connection properties	819
Setting advanced connection properties	820
10.23 Using Kafka as a target	822
Transaction processing by the consumer	823
Prerequisites	824
Limitations and considerations	824
Kafka target data types	825
Setting general connection properties	827
Setting advanced connection properties	835
Working with Confluent Cloud	836
The Replicate envelope	836
Metadata and data messages	838
Using Kerberos authentication on Windows	842
10.24 Using a Log Stream target	845
10.25 Using MapR Streams as a target	847
Transaction processing by the consumer	848
Prerequisites	849
Limitations	849
Supported data types	850
Setting general connection properties	852
Setting advanced connection properties	855
The Replicate envelope	855
Metadata and data messages	857
10.26 Using Microsoft APS PDW as a target	862
Prerequisites	863
Limitations and considerations	863
Security requirements	863
Microsoft APS PDW target data types	863
Setting general connection properties	864
Setting advanced connection properties	865
10.27 Using Microsoft Azure ADLS as a target	867
Microsoft Azure ADLS target overview	868
Prerequisites	868
Limitations and considerations	868
Change data partitioning	869
Data types	870
Setting general connection properties	871
Setting advanced connection properties	876
Generating reference files	879
10.28 Using Microsoft Azure Database for MySQL as a target	881
Prerequisites	882
Limitations and considerations	883
Security requirements	883
Supported data types	884

Setting general connection properties	886
Setting advanced connection properties	887
10.29 Using Microsoft Azure Database for PostgreSQL as a target	889
Prerequisites	890
Limitations	891
Security requirements	891
Microsoft Azure Database for PostgreSQL target data types	891
Setting general connection properties	893
Setting advanced connection properties	894
10.30 Using Microsoft Azure Event Hubs as a target	896
Prerequisites	897
Transaction processing by the consumer	897
Limitations	898
Supported target data types	899
Setting general connection properties	901
Setting advanced connection properties	904
The Replicate envelope	905
Metadata and data messages	907
10.31 Using Microsoft Azure HDInsight as a target	912
Prerequisites	913
Limitations	913
Change data partitioning on Microsoft Azure HDInsight	914
Microsoft Azure HDInsight endpoint target data types	914
Setting general connection properties	915
Setting advanced connection properties	918
10.32 Using Microsoft Azure Synapse Analytics as a target	921
Prerequisites	922
Limitations and considerations	924
Supported data types	925
Setting general connection properties	926
Setting advanced connection properties	929
10.33 Using Microsoft Azure SQL Database as a target	931
Prerequisites	932
Limitations	933
Permissions	933
Microsoft Azure SQL Database target data types	933
Setting general connection properties	935
Setting advanced connection properties	936
10.34 Using Microsoft SQL Server as a target	938
Supported editions	939
Prerequisites	939
Limitation and considerations	940
Permissions	941
Microsoft SQL Server target data types	941
Setting general connection properties	943
Setting advanced connection properties	944
10.35 Using a MySQL-based database as a target	946

Prerequisites	947
Limitations and considerations	948
Permissions	948
Supported data types	949
Setting general connection properties	951
Setting advanced connection properties	952
10.36 Using Netezza Performance Server (NPS) as a target	954
Prerequisites	955
Limitations	955
Permissions	955
Netezza Performance Server (NPS) target data types	956
Setting general connection properties	957
Setting advanced connection properties	958
10.37 Using ODBC to connect to a target	960
Prerequisites	961
ODBC target data types	962
Setting general connection properties	963
Setting advanced connection properties	964
10.38 Using Oracle as a target	966
Supported Oracle database editions	967
Client prerequisites	967
Limitations	968
Security requirements	969
Oracle target data types	969
Setting general connection properties	971
Setting advanced connection properties	973
10.39 Using Pivotal Greenplum as a target	975
An overview of the Pivotal Greenplum target	976
Qlik Replicate Pivotal Greenplum endpoint architecture overview	976
The Pivotal Greenplum Full Load process	978
Applying changes to the Pivotal Greenplum target	978
Prerequisites	978
Provide Pivotal Greenplum account access	979
Security requirements	980
Limitations	980
Pivotal Greenplum data types	980
Setting up the gpfdist program as a service	982
Using multiple gpfdist programs	983
Setting general connection properties	983
Setting advanced connection properties	985
Testing your Pivotal Greenplum Configuration and Environment	986
10.40 Using a PostgreSQL-Based database as a target	990
Prerequisites	991
Limitations	992
Security requirements	992
PostgreSQL Database target data types	992
Setting general connection properties	993
Setting advanced connection properties	995

10.41 Using SAP HANA as a target	997
Prerequisites	998
Limitations and considerations	998
Permissions	998
Supported data types	999
Setting general connection properties	999
Setting advanced connection properties	1000
10.42 Using SAP Sybase ASE as a target	1002
Prerequisites	1003
Limitations and considerations	1003
Security requirements	1003
SAP Sybase ASE database target data types	1003
Non-supported data types	1005
Setting general connection properties	1005
Setting advanced connection properties	1006
10.43 Using SAP Sybase IQ as a target	1008
Prerequisites	1009
Limitations and considerations	1009
Security requirements	1009
SAP Sybase IQ target data types	1009
Setting general connection properties	1010
Setting advanced connection properties	1011
10.44 Using SingleStore as a target	1013
Prerequisites	1014
Limitations and considerations	1014
Permissions	1015
Supported data types	1015
Setting general connection properties	1018
Setting advanced connection properties	1019
10.45 Using Snowflake on AWS as a target	1020
Prerequisites	1021
Limitations and considerations	1022
Required permissions	1022
Snowflake on AWS data types	1023
Setting general connection parameters	1025
Setting advanced connection properties	1027
10.46 Using Snowflake on Azure as a target	1030
Prerequisites	1031
Limitations and considerations	1032
Required permissions	1032
Snowflake on Azure data types	1033
Setting general connection parameters	1035
Setting advanced connection properties	1036
10.47 Using Snowflake on Google as a target	1038
Prerequisites	1039
Limitations and considerations	1039
Required permissions	1040

Snowflake on Google data types	1041
Setting general connection parameters	1043
Setting advanced connection properties	1044
10.48 Using Teradata Database as a target	1046
An overview of the Teradata Database target	1047
Limitations and considerations	1047
Teradata Database target load options	1047
Database availability	1047
Prerequisites	1047
Providing access to the Teradata Database	1049
Security requirements	1050
Teradata Database data types	1050
Setting general connection properties	1053
Setting advanced connection properties	1054
10.49 Using Vertica as a target	1056
Prerequisites	1057
Limitations	1057
Security requirements	1057
Vertica target data types	1058
Setting general connection properties	1059
Setting advanced connection properties	1060
11 Using the Qlik Replicate file channel	1061
11.1 Prerequisites	1062
11.2 Limitations	1062
11.3 Security	1062
11.4 Setting Up Qlik Replicate File Channel tasks	1063
Local task	1063
Remote task	1063
Replicating to multiple targets (distribution)	1064
11.5 Working with the File Channel data files	1065
File-Channel directory structure	1065
11.6 Using the File Channel as a source	1067
11.7 Setting general connection properties	1067
11.8 Setting advanced connection properties	1068
Internal parameters	1068
Settings summary	1069
11.9 Using the File Channel as a target	1069
Setting general connection properties	1069
Setting advanced connection properties	1070
Internal parameters	1070
Settings summary	1071
12 Using the Log Stream	1072
12.1 Terminology	1073
12.2 Overview	1073
The flow	1073
12.3 Step 1: Create a Log Stream target endpoint	1074

12.4 Step 2: Create and run a Log Stream Staging task	1075
Log Stream Staging limitations and considerations	1078
12.5 Step 3: Create a replication task to transfer the staged data to the target	1078
Duplicate the source endpoint and add it to the log stream staging task	1079
Create the replication task using the duplicated source endpoint	1079
Replication task limitations and considerations	1081
13 Customizing tasks	1082
13.1 Table Settings	1082
Performing General tasks for a single table/view	1083
Defining transformations for a single table/view	1084
Using filters	1092
Parallel Load	1098
Handling LOB columns	1103
Message format	1106
Full Load	1106
13.2 Defining global rules	1107
Starting the Global Transformation Rules wizard	1107
.....	1118
.....	1118
Starting the Global Filter Rules wizard	1131
Managing global rules	1136
13.3 Using the Expression Builder	1137
Overview of the Expression Builder	1137
Build an expression	1139
Parse an expression	1140
Test an expression	1140
Using elements in the Expression Builder	1142
13.4 Task Settings	1165
Metadata	1166
Bidirectional	1171
Full Load	1171
Change Processing	1174
Error handling	1186
Logging	1192
Character substitution	1194
File uploads	1195
Message Format	1196
Transformations and Filters	1201
More options	1202
14 Working with tasks at runtime	1203
14.1 Running a task	1203
How to run a task	1203
Using the Run button options	1204
Advanced Run Options	1205
Recovering from data folder loss or corruption	1210
14.2 Viewing the task status	1212
14.3 Accessing task messages and notifications	1212

Viewing Notifications	1213
View Log Messages for a task	1214
15 Monitoring and controlling replication tasks	1216
15.1 Viewing information in the monitor	1216
15.2 Monitoring full-load operations	1216
General information for a Full Load	1216
Detailed information for the Full Load	1217
Monitoring throughput in a Full Load operation	1224
15.3 Monitoring change processing operations	1224
General Change Processing information	1224
Detailed Change Processing information	1226
15.4 Viewing messages	1233
15.5 Using the monitor tools	1233
Viewing history information	1234
Setting the task logging level	1234
Viewing, downloading, and manually rolling over task log files	1235
Deleting log files	1236
Deleting old partitions manually	1237
Downloading a memory report	1237
Downloading a diagnostics package	1237
16 Qlik Replicate server settings	1239
16.1 Notifications settings	1239
Defining notifications	1240
Define the action that triggers the notification	1241
Define which changes of status trigger the notification	1242
Define errors or warnings that trigger the notification	1244
Define the recipients	1245
Define the notification message	1245
Associate tasks with the notification	1250
Review the notification rule	1251
Define the event that triggers the notification	1251
Review the notification rule	1254
Setting up Mail parameters	1257
Creating a Default Recipient List	1258
Replicate Event IDs in Windows Event Log	1259
16.2 License settings	1260
Requesting a license	1260
Registering a license	1263
Viewing a license	1264
16.3 Global error handling	1264
16.4 Logging	1265
Setting logging levels for the Server and File Transfer Service	1266
Setting automatic roll over and cleanup	1267
Viewing and downloading log files	1268
Manually rolling over the log files	1269
Deleting server, task and FTS log files	1269
16.5 File Transfer Service	1269

How it works	1269
Compression	1270
Encryption	1270
Defining a File Transfer Service	1270
Editing a File Transfer Service	1271
Deleting a File Transfer Service	1271
16.6 Scheduling jobs	1272
16.7 User permissions	1274
Managing user permissions	1275
16.8 Resource control	1276
Disk space	1276
System memory	1277
16.9 External utilities	1277
16.10 Endpoints	1278
Endpoint Servers	1278
Default proxy server	1280
16.11 Transformations and Filters	1281
16.12 More options	1282
More options	1282
A Using Change Tables	1283
A.1 Working with Change Tables	1283
Handling truncate operations	1283
Using the Change Table model	1284
A.2 Change Tables	1284
A.3 Usage example	1287
B Using an Audit table	1289
C Creating dump files	1292
D Setting up Qlik Replicate in a cluster environment	1293
D.1 Setting up Qlik Replicate in a Windows Server cluster (HA)	1293
Step 1: Install Qlik Replicate in the cluster	1293
Step 2: Add the Qlik Replicate services	1295
Step 3: Define the dependencies for each service	1295
Step 4: Enable different console addresses in a high availability environment	1296
D.2 Setting up Qlik Replicate in a Linux cluster	1297
E Control Tables	1298
E.1 Apply exceptions	1298
E.2 Replication status	1299
E.3 Suspended Tables	1300
E.4 Replication history	1300
E.5 Change data partitions	1301
E.6 DDL history	1302
F Using HP NonStop SQL/MP as an ODBC target	1305
F.1 Prerequisites	1305
F.2 Table settings	1306
F.3 Task setting limitations	1307

G Impact of DST change on Qlik Replicate	1308
H Metadata file description	1310
I Supported Platforms and Endpoints	1313
I.1 Supported platforms	1313
Supported Windows platforms	1313
Supported Linux platforms	1314
I.2 Supported source endpoints	1314
Cloud-based	1314
ARC-based	1316
AIS-based	1316
File	1317
Data Lake	1317
Relational databases	1317
Data warehouses	1318
ODBC	1318
NoSQL	1319
Other	1319
	1320
I.3 Supported target endpoints	1320
Cloud-based	1320
Streaming	1324
File-based	1324
Hadoop	1325
Data warehouses	1325
Relational databases	1325
ODBC	1326
I.4 Endpoints supported in bidirectional replication	1326
I.5 Compatibility with related products	1327
I.6 Supported browsers	1327
J Replicate Loggers	1328
J.1 ADDONS	1328
J.2 ASSERTION	1329
J.3 COMMON	1329
J.4 COMMUNICATION	1329
J.5 DATA_RECORD	1329
Example	1329
J.6 DATA_STRUCTURE	1329
J.7 FILE_FACTORY	1330
J.8 FILE_TRANSFER (AKA CIFTA)	1330
J.9 INFRASTRUCTURE	1330
J.10 IO	1330
Example:	1330
J.11 METADATA_CHANGES	1330
J.12 METADATA_MANAGER	1330
J.13 PERFORMANCE	1330
J.14 REST_SERVER	1330


J.15 SERVER	1331
J.16 SORTER	1331
J.17 SORTER_STORAGE	1331
J.18 SOURCE_CAPTURE	1331
J.19 SOURCE_LOG_DUMP	1331
J.20 SOURCE_UNLOAD	1331
J.21 STREAM	1332
J.22 STREAM_COMPONENT	1332
Example	1332
J.23 TABLES_MANAGER	1332
J.24 TARGET_APPLY	1332
J.25 TARGET_LOAD	1332
J.26 TASK_MANAGER	1332
J.27 TRANSFORMATION	1333
Example:	1333
J.28 UTILITIES	1333
K Replicate add-ons API	1334
K.1 Memory management methods	1334
create_pool	1334
destroy_pool	1335
clear_pool	1335
calloc	1335
get_ctx_pool	1336
get_ctx	1336
set_ctx	1336
K.2 Log management methods	1337
log_error	1337
log_warning	1337
log_trace	1338
K Kafka Avro consumers API	1339
K.3 Decoding a self-describing message	1340
K.4 Decoding a message by referenced schema ID	1340
K.5 Typical consumer logic	1341
K.6 Classes and Methods	1341
AttunityMessageDecoder	1341
AttunityMetadataMessage	1342
AttunityDataMessage	1343
AttunityTableLineage	1343
AttunityTableColumn	1343
AttunityDataType	1344
AttunityDataHeaders	1344
AttunityDecoderException	1345
K.7 Sample Code Snippet	1346
L Glossary	1348

1 What's new?

This section describes the new and enhanced features from Replicate May 2023 initial release until and including Replicate May 2023 SR01.



In addition to these release notes, customers who are not upgrading from the latest GA version (Replicate May 2023) are advised to review the release notes for all versions released since their current version.

Customers should also review the Replicate release notes in  [Qlik Community](#) for information about the following:

- Migration and upgrade
- End of life/support features
- Newly supported versions and third-party software
- Resolved issues
- Known issues

1.1 New endpoints and endpoint enhancements in Replicate May 2023 service release 01

This section describes the new and enhanced endpoint features.

New Confluent Cloud target endpoint

This version introduces support for replicating data from any supported source to Confluent Cloud.

Using Confluent Cloud as a target (page 660)

New authentication option for Kafka target

When publishing to the Confluent Schema Registry, you can now choose to authenticate using both a certificate and a user name and password.

Schema Registry connection properties (page 832)

1.2 Endpoints and endpoint enhancements first introduced in Replicate May 2023 initial release

New Google Cloud Pub/Sub target endpoint

This version introduces support for replicating data from any supported source to Google Cloud Pub/Sub.

Using Google Cloud Pub/Sub as a target (page 722)

New IBM DB2 for z/OS target endpoint

This version introduces support for replicating data from any supported source to IBM DB2 for z/OS.

Using IBM DB2 for z/OS as a target (page 813)

New Google Cloud AlloyDB for PostgreSQL endpoints

Google Cloud AlloyDB for PostgreSQL is now supported as a source endpoint and as a target endpoint.

Using Google Cloud AlloyDB for PostgreSQL as a source (page 211)

Using Google Cloud AlloyDB for PostgreSQL as a target (page 709)

SAP ODP source endpoint enhancements

SLT support

This version introduces support for replicating data from a SAP Landscape Transformation Replication Server. To facilitate this functionality, two new options have been added to the **Advanced** tab of the SAP ODP endpoint settings:

- A new ODP context has been added: **SAP LT Replication (SLT)**
- **SLT alias:** This field is shown when SAP LT Replication (SLT) is selected and must match the alias defined in the SAP configuration created in SAP LT Replication Server Cockpit (transaction LTRC).

Delta processing enhancements

This version introduces greater control over delta processing. To facilitate this, the following options have been added to the Advanced tab of the SAP ODP endpoint settings:

- **History data** - This is the default mode. When this mode is selected, all data will be applied as INSERTS, thereby preserving previous record versions. When working in this mode, the following options are also available:
 - **Apply original primary key** - Retrieve the primary key settings from the ODP metadata.
 - **Reverse summable fields** - Reverse the values of summable fields. Before-image values have reversed signs. Enabling this setting will restore the sign to its prior state and recalculate the original value.
- **Current data** - When this mode is selected, the actual change operation (INSERT, UPDATE, or DELETE) will be performed on the target.

Using SAP ODP as a source (page 549)

IBM DB2 for z/OS source endpoint - data server client support

In previous versions, customers running Replicate on Linux needed to install the full ODBC client package to work with the IBM DB2 for z/OS source endpoint. From this version, installing just the data server client (which requires less disk space) is also supported.

Using IBM DB2 for z/OS as a source (page 276)

AWS Aurora Cloud for PostgreSQL source endpoint: non-superuser support

It is now possible to work with the AWS Aurora Cloud for PostgreSQL source endpoint without the superuser role.

Using an account without the "superuser" role (page 134)

Log Stream tasks

Log Stream tasks now support the source change position (e.g. SCN or LSN) **Advanced Run** option with the Oracle source endpoint.

Using the Log Stream (page 1072)

Endpoint proxy server enhancements

From this version, you can now choose whether to use the default proxy server configured in the server settings, or provide proxy settings specific to the endpoint. Supported with the following endpoints only:

- Databricks Lakehouse (Delta)
- Databricks Cloud Storage
- Google Cloud Big Query
- Amazon Redshift
- Microsoft Azure Synapse Analytics

See also: *Default proxy server (page 30)*

Microsoft Azure Data Lake Storage (ADLS) Gen2

For target endpoints that offer the option of using Microsoft Azure Data Lake Storage (ADLS) Gen2 storage, it is now possible to choose whether to use a proxy server to access the **Staging storage** and **Azure Active Directory** or either one of them.

Amazon Redshift target endpoint - AWS PrivateLink support

Customers using the Amazon Redshift target endpoint can use AWS PrivateLink to connect to a virtual private cloud (VPC). To enable this functionality, a **Use AWS PrivateLink** option and **VPC Endpoint URL** field have been added to the **General** tab of the Amazon Redshift target endpoint settings.

Using Amazon Redshift as a target (page 625)

Support for Unity Catalog with Databricks endpoints

The version introduces support for Unity Catalog when using the Databricks Cloud Storage or Databricks Lakehouse (Delta) target endpoints.

Using Databricks Lakehouse (Delta) as a target (page 671)

Using Databricks (Cloud Storage) as a target (page 682)

Data type enhancements

Change to parquet data type mapping

The data type mapping for Parquet format has been changed for the following endpoints:

- Amazon S3
- Microsoft Azure ADLS
- Google Cloud Storage

The BYTES data type, which was previously mapped to FIXED_LEN_BYTE_ARRAY (\$LENGTH) will now be mapped to BYTE_ARRAY.



This change only affects newly created endpoints. Existing endpoints will continue to use the old mapping.

Multirange data type support with PostgreSQL-based source endpoints

The following multirange data types are now supported with all PostgreSQL-based source endpoints.

- INT4MULTIRANGE
- INT8MULTIRANGE
- NUMMULTIRANGE
- TSMULTIRANGE

Extended Parallel Load support

Data can now be loaded to the **Kafka** and **Amazon MSK** target endpoints using Parallel Load. In Full Load replication mode, you can use Parallel Load to accelerate the replication of large tables by splitting the table into segments and loading the segments in parallel. Tables can be segmented by data ranges, by partitions, or by sub-partitions.

Parallel Load (page 1098)

Default quote character in file-based endpoints

New endpoints will be created with double-quotes (") as the default quote character instead of an empty value. The change applies to the following target endpoints: Amazon S3, File, Microsoft ADLS, and Google Cloud Storage.

Renamed IBM Netezza target endpoint

The IBM Netezza target endpoint has been renamed to Netezza Performance Server (NPS).

Microsoft Azure SQL (MS-CDC) endpoint - Geo Replica support

This version introduces support for reading events from Geo Replica.

Oracle HSM (Hardware Security Module) support

This version introduces support for reading data from both encrypted tablespaces and encrypted columns during CDC. This is true for both Replicate Log Reader and Oracle LogMiner to access the redo logs.

Supported encryption methods (page 426)

1.3 What's new on the server side?

This section describes the new and enhanced server-side features.

Default proxy server

Instead of needing to configure a proxy server for each endpoint, you can now configure a default proxy server which can be used by all endpoints. To facilitate this functionality, a new **Default proxy server** section has been added to the server settings' **Endpoints** tab. Supported with the following endpoints only:

- Databricks Lakehouse (Delta)
- Databricks Cloud Storage
- Google Cloud Big Query
- Amazon Redshift
- Microsoft Azure Synapse Analytics

Endpoints (page 1278)

Support special characters in column names used in expressions

The new **Support special characters in column names used in expressions** option can be set globally for all tasks or individually for a specific task. The option is located in the server and task settings' **Transformations and Filters** tab. Enable the option to include source column names with special characters in expressions defined for this task. An example of such a column name would be **special#column**.



Before enabling this option, make sure to read the full description of this feature in the associated help topic. Failure to do so might result in data corruption.

Transformations and Filters (page 1281)

2 Introduction

This section describes the main concepts of data replication and the major components of Qlik Replicate.



The term "endpoint" is used generically throughout this guide to refer to a data repository that can be used as a source and/or target in a Qlik Replicate task. Examples of such repositories include relational databases (such as Oracle) and files.

In this section:

- *What you need (page 97)*
- *Replication Explained (page 34)*
- *Full Load and CDC processes (page 35)*
- *Replication topologies (page 36)*
- *Limitations and considerations (page 36)*

2.1 What is Qlik Replicate?

Qlik Replicate is a simple, powerful, easy-to-implement solution that provides replication between various endpoints. Replicate lets you:

- Load data efficiently and quickly to operational data stores/warehouses
- Create copies of production endpoints
- Distribute data across endpoints

Replicate is designed to scale and support large scale enterprise data replication scenarios with a scalable multi-server, multi-task, and multi-threaded architecture.

Replicate consists of a Web-based console and a replication server to replicate data across both heterogeneous and homogeneous data sources. It provides users with instant visibility into current and historical tasks, status, performance, and resource usage information.

Replicate can execute replication tasks between a large number of enterprise endpoints including Oracle, Microsoft SQL Server, and IBM DB2. It uses a "Click-2-Replicate" design that simplifies the replication process by automating the steps required to build a replication solution.

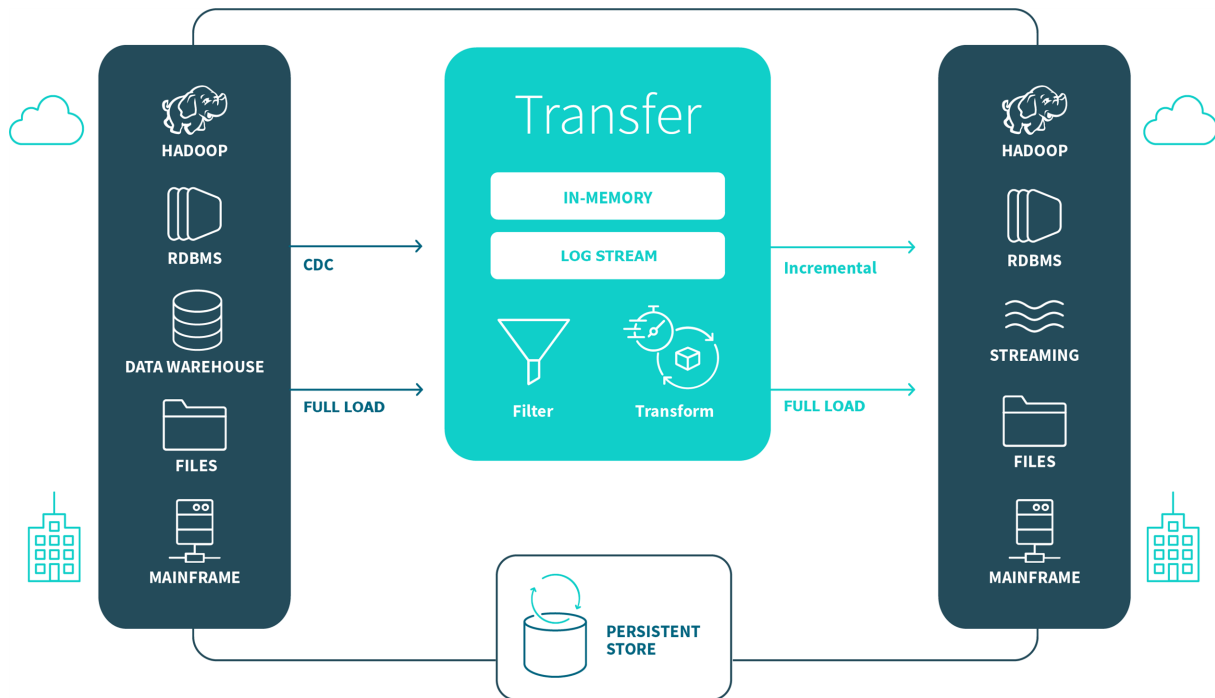
When you set up a task in Replicate, you can select any of the following:

- **Full Load Replication:** Creates files or tables on the target endpoint, automatically defines the metadata that is required on the target, and populates the tables with data from the source
- **Change Processing, also called Change Data Capture (CDC):** Captures changes in the source data or metadata as they occur and applies them to the target endpoint as soon as possible, in near-real time
- **Store Changes:** Captures changes in the source data or metadata as they occur but does not apply them to the target

Replication is log based, which means that it reads only the changes. This reduces the impact on the source endpoints.

System architecture

The following diagram shows the basic architecture of Qlik Replicate.



In the initial load process, Qlik Replicate reads a filtered stream of rows (with relevant columns only) and passes them to the transformation process for further filtering and subsequent writing to the target endpoint (in the expected output format).

The CDC process obtains a stream of filtered events or changes in data or metadata from the transaction log file. It then buffers all changes for a given transaction into a single unit before forwarding them to the target when the transaction is committed. During the initial load process, CDC also buffers all changes that occur within a transaction until all affected tables have been loaded.

2.2 Terminology

The following section describes some key terms used throughout this Help.

Change Data Capture (CDC)

Captures changes in the source data or metadata as they occur and applies them to the target endpoint as soon as possible, in near-real-time. The changes are captured and applied as units of single committed transactions and several different target tables may be updated as the result of a single source commit. This guarantees transactional integrity in the target endpoint. The CDC process for any file or table starts as soon as the data loading operation for the file or table begins.

Full load

Creates all defined files or tables on the target endpoint, automatically defines the metadata that is required at the target, and populates the tables with data from the source.

Apply latency

The gap in seconds between capturing a change in one of the source tables and applying that change to the target endpoint.

Latency when applying large transactions

This is best explained by way of example. When the most recent Apply Latency value was 10 seconds and now a transaction of one million rows gets committed at the source endpoint, Replicate starts to apply that transaction to the selected target and it will take some time to write all the changes to the target (for example 60 seconds). During the next 60 seconds, the latency value gradually grows to 70 seconds for the last change in the transaction. Once the transaction is committed, the latency drops back to the 'regular' latency (10 seconds in this case).

Latency when no transactions are being applied

When a time period passes with no changes applied to the target, the latency calculation is based on the time difference between the current time and the timestamp of the last change event read from the transaction log. This could happen, for example, if there is a high volume of activity on tables that were not selected for replication in the current task.

Source latency

The gap in seconds between when the source database wrote an event to its transaction log and when Replicate captured that change.

Target latency

The gap between when a commit is seen by Replicate (reading the source transaction log) and when the changes of that commit are seen in the target.

Overall latency

The overall latency is defined as the time gap between when a change is committed in the source database and when it is visible in the target database.

Source endpoint

A collection of files or tables managed by an endpoint management system (such as, Oracle, SQL Server) that is part of the main computing service of the IT organization of an enterprise. This source continuously updated, may need to provide a high throughput rate, may have strict 24/7 up-time requirements, and may reference or update a number of tables in the course of a single logical transaction while providing transactional consistency and integrity for the data.

Target endpoint

A collection of files or tables managed by an Endpoint Management System (DBMS), which may be different from the DBMS managing the source endpoint. It contains data that is derived from the source. It may contain only a subset of the tables, columns, or rows that appear in the source. Its tables may contain columns that do not appear in the source but are transformations or computations based on the source data.

Net Changes table

Replicate performs data replication based on changes that appear in the source database's transaction log. A single update operation on the source, such as "UPDATE MyTable SET f1=..., f2=..." could potentially update many rows in the source database and create a large number of change records that Replicate will need to apply to the target. Replicate offers two Change Processing modes: **Transactional apply** and **Batch optimized apply**. In **Transactional apply** Change Processing mode, Replicate essentially applies each change to the target, which may take much longer than the original UPDATE took on the source. **Batch optimized apply** mode, on the other hand, is designed to handle efficient replication of a large number of changes. In this mode, Replicate accumulates changes for multiple tables in a memory cache. Repeated changes to the same row are updated in the memory cache. When the maximum memory cache size defined for the task is reached (or when the configured time has elapsed), Replicate does the following:

- a. Writes the cached (net) changes to a special table on the target (the **Net Changes** table),
- b. Bulk uploads the changes to the target table
- c. Uses SQL statements to update the target tables based on the data in the **Net Changes** table.



*Note that for Oracle, Replicate uses a **Net Changes** table per each source table with changes, while for other source endpoints a single **Net Changes** table is used.*

2.3 Replication Explained

Replication is a process that keeps two or more collections of computerized information identically synchronized. It facilitates:

- **Load reduction:** Keeping a complete or partial copy of a collection on a different server reduces the load on the main server.
- **Improved service:** Accessing a copy of the data can provide better service to users than having them access the original data.
- **Restricted data access:** If some users should only have access to a subset of data, replicating only part of a collection makes it easy to enforce security restrictions.
- **Geographic distribution:** Making only a subset of data relevant to a specific node (or location) available is beneficial in widely distributed enterprises (such as a chain of retail stores or warehouses). You can still make all data available at a central location for less frequent use.
- **Disaster Recovery:** Keeping a copy of the main data available allows for setting up rapid fail-over clusters (the capability to switch over to a redundant or standby computer server in case the main system fails).

- **"Cloud" computing:** Replicating data allows for implementing what is commonly known as cloud computing (the on-demand storage, management, and processing of Internet-based data).

The information replicated is stored as files or in a database. In the case of files, the structure and content of a file are known only to the specialized programs that use the file. Databases are managed by database management systems (DBMS) that make use of standardized descriptions of the structure of the information (such as tables, columns, rows, and data types). These descriptions are known collectively as metadata and allow a general-purpose replicator to carry out relevant operations (for example filtering and data transformations) without the need to know anything about the contents or “meaning” of the data. Because file systems do not contain metadata, operations available for replication are more limited.

During replication, a collection of data is copied from system A to system B, where A is known as the source (for this collection) and B is known as the target. A system can be a source, a target, or both (with certain restrictions). A complex replication topology has a number of sources, targets, and data collections defined.

The replication process must account for the fact that source data may be changing while being copied. It is not possible to make or maintain copies instantaneously and to stop the source computer to “freeze” the information. Therefore, replication must account for:

- **Integrity:** The target data must reflect the complete result of all changes made to the source data during the replication process.
- **Consistency:** If a change affects different tables, rows, or files, the copy must reflect these changes consistently across all affected tables, rows, or files.
- **Latency:** The replication process must aim at keeping latency at a minimum. Ideally, it should not exceed a few seconds.

2.4 Full Load and CDC processes

The full load process creates files or tables at the target endpoint, automatically defines the metadata that is required at the target, and populates the tables with data from the source. Unlike the CDC process, the full load process loads the data one entire table or file at a time, for maximum efficiency.

The source tables may be subject to update activity during the Load process. However, there is no need to stop processing in the source. Replicate automatically starts the CDC process as soon as the load process starts. It does not apply the changes to the target until after the load of a table completes because the data on the target might not be consistent while the load process is active. At the conclusion of the load process, however, Replicate guarantees consistency and integrity of the target data.

If the load process is interrupted, it continues from wherever it stopped when restarted.

You can add new tables to an existing target without reloading the existing tables. Similarly, you can add or drop columns in previously populated target tables without reloading.

The CDC process captures changes in the source data or metadata as they occur and applies them to the target endpoint as soon as possible in near real time. It captures and applies the changes as units of single committed transactions and can update several different target tables as the result of a single source commit. This guarantees transactional integrity in the target endpoint. The CDC process for any file or table starts as soon as the data load process for the file or table begins.

CDC operates by reading the recovery log file of the source endpoint management system and grouping together the entries for each transaction. The process employs techniques that ensure efficiency without seriously impacting the latency of the target data. If the CDC process cannot apply the changes to the target within a reasonable amount of time (for example when the target is not accessible), it buffers the changes on the Replication server for as long as necessary. There is no need to re-read the source DBMS logs, which may take a long time.

2.5 Replication topologies

Qlik Replicate can be set up to work in different topologies including one to one, logical independence, and hub and spoke. The following topic provides a brief overview of these topologies.

One to one

In a one-one topology, there is one source and one target endpoint. When the source and target endpoints are distinct, Qlik Replicate guarantees transactional integrity and consistency. If you use two different replication tasks, the endpoints may switch roles, allowing two-way synchronization.



If the same row in a table is updated by two different replication tasks, the result of two-way synchronization may be unpredictable. A problem can occur even if two different rows are referentially related, that is if some application updates a row based on reading a value in a different row. If the rows are updated concurrently on the source and the target, the result may be unpredictable. (CDC has no way of knowing exactly when a row was read by an application on one system relative to its having been changed on another system. Read operations are typically not logged.) Such occurrences are rare, but they can occur.

Logical independence

Two-way replication works best when updates of a row on a source and on a target are entirely autonomous and do not affect each other. There is an assumption that any table or a horizontal or vertical segment of a partitioned table can only be updated in one source. Qlik Replicate allows updating the same row in several places, but in this case, the columns being updated must be distinct. Another assumption is that if a data value in one row depends on or is derived from a value in another row, the values can be changed only on the same server but nowhere else (except by the Replicator). This is called logical independence. With logical independence, concurrent update conflicts cannot occur during replication.

Hub and spoke

Many-to-one and one-to-many relationships can be combined into a hub-and-spoke topology, which allows the merging of data into multiple targets and then distributing to other targets. It does not allow cycles or multiple paths for propagating changes. The hub-and-spoke topology is that of an acyclic directed graph.

2.6 Limitations and considerations

When working with Qlik Replicate, you should be aware of the following limitations and considerations:

- It is not recommended to performing database maintenance tasks during replication as doing so might result in unpredictable behavior.
- Replicate does not support replication of Primary Keys that are LOB data types.
- When replicating a table that has no Primary Key or Unique Index, LOB columns will not be replicated.
- When the [Limit LOB size to](#) option is enabled, replication of structured data LOBs (e.g. XML, JSON, IMAGE, etc.) may truncate (and thereby invalidate) the structured data in the target LOB.
- In Batch optimized apply mode, if the target table has more columns than the source table, any values in the extra columns will be replaced with NULL.
The workaround is to create two tasks. One task for the target table(s) with extra columns and the other task for the source table(s) which have the same number of columns as the target tables. Then, run the task for the target table(s) with extra columns in [Transactional apply](#) mode and run the other task (where the target tables do not have extra columns) in [Batch optimized apply](#) mode. Note, however, that updating large tables in Transactional apply mode may impact performance.
- When Replicate creates a new table in the target endpoint, it defines only *one* index on the table. The index will either be the Primary Key or the first Unique Key (according to alphabetical order) of the table. No other indexes will be defined in the target. If additional indexes are required, these will need to be defined manually.
- If a Unique Index/Primary Key in any of the source tables contains NULL values in multiple rows, UPDATE and DELETE operations on *one* of the rows will UPDATE /DELETE *all* of the target rows (in the Unique Index/Primary Key) that have a NULL value.
- LOB columns are always created as nullable on the target database. If you create the target table(s) manually, then you must set all LOB columns to nullable.
- If you stop a task after Full Load completes, make some changes to the source tables, and later resume the task from timestamp (by selecting the **Start processing changes from** run option), some changes may not be replicated to the target. This usually only happens if the transaction logs in the source database have been deleted due to a log purge policy. In this case, Replicate will resume the task from the last change in the current transaction log.
- When replicating tables without a Primary Key, there is no way to verify whether a record already exists on the target. This may result in data inconsistency when UPDATE and DELETE operations are performed on the target database.
- Replication of calculated values is not supported during Change Processing.
- If a task fails with a recoverable error on the target while it is starting, it will not read changes from the source.
- Cached changes may be duplicated in a target table that does not have a Unique Index.
- A unique index consisting of several ascending and descending columns will always be replicated to the target as ascending columns. In other words, the descending columns will become ascending columns.
- When the source table contains an identity column, Replicate does not create the identity column on the target table. In this case, the table will need to be created manually on the target endpoint.
- Replication of tables with the same name as any of the Replicate Control tables is not supported. For a list of the Control Table names, see *Control Tables (page 1298)*.
- CREATE TABLE operations performed on the source while a task is stopped will be applied to the target when the task is resumed, but will not be recorded as a DDL in the [attrep_ddl_history Control Table](#).

- Due to access restrictions on Windows registry entries, two installations of Replicate *on the same machine* must be installed under the same user account. If they are installed under different accounts, only one of the installations will work.
- When using [filters](#) that do not contain a Primary Key, DELETE operations will not be captured from endpoints that do not support Before-images. Examples of such endpoints include all PostgreSQL-based sources, Salesforce source, and Oracle source that does not have full supplemental logging on columns included in the filtering condition.
- Replication of source tables that contain two columns with the same name, but in different cases (e.g. ADDRESS and address), is not supported.
- Replicate does not capture DDLs of type ALTER TABLE that change a column type from VARCHAR to CHAR. However any change to the size will be captured. Additionally, Replicate will only capture DDLs of type ALTER TABLE that change a column type from CHAR to VARCHAR if the size has changed.

Example:

- Change VARCHAR(20) to CHAR(20) - No change in target
- Change VARCHAR(20) to CHAR(55) - Target will be VARCHAR(55)
- Change CHAR(20) to VARCHAR(20) - No change in target
- Change CHAR(20) to VARCHAR(55) - Target will be VARCHAR(55)

3 Installing Qlik Replicate

This section describes how to install, upgrade, and uninstall Qlik Replicate.

In this section:

- *Recommended hardware configuration (page 39)*
- *Setting up Qlik Replicate on Windows (page 40)*
- *Setting up Qlik Replicate on Linux (page 49)*

3.1 Recommended hardware configuration

This section describes the recommended hardware configurations for using Qlik Replicate. For information on the software requirements for using Qlik Replicate, see *Installing Qlik Replicate (page 39)*.

The following table describes the recommended hardware configuration for installing Qlik Replicate on Windows and Linux operating systems. Note that the recommendations apply to mid-scale systems (i.e. hundreds of tasks) rather than large-scale systems (i.e. thousands of tasks).

Recommended hardware configurations

	Basic System	Large System	Extra-Large System	Notes:
Processor	Quad core	Quad core base Dual-core per task	8-core base Quad core per task	Additional cores are useful in any of the following situations: <ul style="list-style-type: none"> • Many tasks running in parallel • Full-load performance priority • Multiple full-load processes running in parallel
Memory	8 GB	32 GB	64 GB	More memory is useful in any of the following situations: <ul style="list-style-type: none"> • Many tasks running in parallel • Long-running transactions on the source endpoint (for example, monthly batch processing) • Many active users on the source system

	Basic System	Large System	Extra-Large System	Notes:
Disk requirements (SSD is recommended for optimal performance)	320 GB	500 GB RAID	500 GB RAID	<p>A faster disk is useful in any of the following situations:</p> <ul style="list-style-type: none"> Using a file-based target, such as Pivotal Greenplum or Actian Vector Long-running source transactions that may not fit into memory Using tasks that are set up to continue processing during target outage <p>A larger disk is required in any of the following situations:</p> <ul style="list-style-type: none"> Using tasks that are set up to continue processing during target outage Very large source transactions that do not fit into memory <p>RAID is recommended for system recovery in case of disk failure for all configurations.</p>
Network	1 Gbps	10 Gbps	Two 10 Gbps	-

3.2 Setting up Qlik Replicate on Windows

This section describes how to install Qlik Replicate on Windows and contains the following topics:

- *Windows permissions (page 41)*
- *Windows software requirements (page 44)*
- *Installing or upgrading Qlik Replicate on Windows (page 44)*
- *Starting and stopping the Qlik Replicate Server service on Windows (page 46)*
- *Silently installing Qlik Replicate (page 46)*
- *Silently upgrading Qlik Replicate (page 47)*
- *Silently uninstalling Qlik Replicate (page 48)*
- *Changing the data directory location on Windows (page 49)*

Windows permissions

By default, Qlik Replicate must be installed as an Administrator. After installing Qlik Replicate, you can opt to run Replicate under a different account, as described in *Running Replicate under a different account* (page 42).

The following privileges are required to start the **Qlik Replicate UI Server** service (which is run as a local system service), but are dropped as soon as the service is started:

- SE_CREATE_GLOBAL_NAME
- SE_CREATE_PAGEFILE_NAME
- SE_CREATE_PERMANENT_NAME
- SE_CREATE_SYMBOLIC_LINK_NAME
- SE_CREATE_TOKEN_NAME
- SE_DEBUG_NAME
- SE_ENABLE_DELEGATION_NAME
- SE_IMPERSONATE_NAME
- SE_INC_BASE_PRIORITY_NAME
- SE_INCREASE_QUOTA_NAME
- SE_INC_WORKING_SET_NAME
- SE_LOAD_DRIVER_NAME
- SE_LOCK_MEMORY_NAME
- SE_MACHINE_ACCOUNT_NAME
- SE_MANAGE_VOLUME_NAME
- SE_PROF_SINGLE_PROCESS_NAME
- SE_RELABEL_NAME
- SE_REMOTE_SHUTDOWN_NAME
- SE_RESTORE_NAME
- SE_SECURITY_NAME
- SE_SHUTDOWN_NAME
- SE_SYNC_AGENT_NAME
- SE_SYSTEM_ENVIRONMENT_NAME
- SE_SYSTEM_PROFILE_NAME
- SE_SYSTEMTIME_NAME
- SE_TAKE_OWNERSHIP_NAME
- SE_TCB_NAME
- SE_TIME_ZONE_NAME
- SE_TRUSTED_CREDMAN_ACCESS_NAME
- SE_UNDOCK_NAME

In addition, the account that runs Replicate needs to be granted access to the **Data** directory (~\Qlik\Replicate\Data) as well as any directory containing files (such as CSV files) that need to be used in a replication task.

Running Replicate under a different account

You can run Replicate under an account with less privileges than the default account. Both the Qlik Replicate Server service and the Qlik Replicate UI Server service can be run under different accounts, as described in the following sections.



- *The Salesforce and MongoDB source endpoints are not supported when Replicate is run under a non-admin account.*
- *As some endpoints may require stronger privileges than others, running Replicate under an account with insufficient privileges may cause replication issues.*

Changing the Qlik Replicate Server Service account

Changing the default Qlik Replicate Server Service Log On user for admin users

1. Add the user to the "Log on as a service" policy in the "Local Security Policy".
2. Edit the **Qlik Replicate Server** service to log on with the user.
3. Restart the Qlik Replicate Server service.
4. Open Task Manager and verify that the **repctl.exe** process is running under the specified user.

Changing the default Qlik Replicate Server Service Log On user for non-admin users



If you already performed this procedure in an earlier version of Replicate, you only need to perform the additional steps described in Steps required after upgrading to Replicate November 2022 or later (page 43) below.

1. Add the user to the "Log on as a service" policy in the "Local Security Policy".
2. Set the user as the Owner of the **log.key**, **mk.dat**, and **muk.dat** files.
The default location of these files is:
`C:\Program Files\Attunity\Replicate\data`
3. Grant the user full control over the **log.key**, **mk.dat**, and **muk.dat** files.
4. Grant the user full control over the following registry entries:
`HKEY_LOCAL_MACHINE\SOFTWARE\Qlik\Qlik Replicate`
`HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\Windows Error Reporting\LocalDumps\Repctl.exe`
5. Grant the user full control over the Replicate jvm folder
The default location of this folder is:
`C:\Program Files\Attunity\Replicate\jvm`
6. For both the <INSTALL-DIR>\Attunity**Replicate** and <INSTALL-DIR>\Attunity\Replicate**data** folders, open the folder properties and set the folder permissions as follows:
 - a. Grant the user full control.
 - b. Click the **Advanced** button.
 - c. In the **Permissions** tab, click the **Change permissions** button.
 - d. Select **Replace all child object permission entries with inheritable permission entries from this object** and click **OK**.
7. Grant the user write permission on <INSTALL-DIR>\Attunity\Replicate\endpoint_srv\bin.

8. If you have not yet registered your Replicate license, register it now.
9. Verify that the **muk.dat** file exists in <INSTALL-DIR>\Attunity\Replicate\endpoint_srv\data and then grant the user full control over this folder.
10. Grant the user full control over any target folders defined in relevant target endpoints (for example, the **Target folder** specified in the File target endpoint settings).
11. If execution of [user-defined commands is allowed](#), add the user to the "Replace a process level token" policy in "Local Security Policy".
12. Edit the **Qlik Replicate Server** service to log on with the specified user.
13. Stop the **Qlik Replicate UI Server** service.
14. Restart the **Qlik Replicate Server** service.
15. Start the **Qlik Replicate UI Server** service.
16. Open Task Manager and verify that the **repctl.exe** process is running under the specified user.

Steps required after upgrading to Replicate November 2022 or later

If you already performed the *Changing the default Qlik Replicate Server Service Log On user for non-admin users* (page 42) procedure in an earlier version of Replicate, after upgrading to Replicate November 2022, you need to perform the following additional steps:

1. Edit the **Qlik Replicate Server** service to log on with the user you added to the "Log on as a service" policy (in the "Local Security Policy").
2. Grant the user full control over the following registry entry:
HKEY_LOCAL_MACHINE\SOFTWARE\Qlik\Qlik Replicate
3. Stop the **Qlik Replicate UI Server** service.
4. Restart the **Qlik Replicate Server** service.
5. Start the **Qlik Replicate UI Server** service.

Changing the Qlik Replicate UI Server Service Account



The user can either be an admin user or a non-admin user. If you change the user to a non-admin user, the user must be the same as the non-admin user set in Changing the Qlik Replicate Server Service account (page 42) above (assuming that a non-admin user was set).

To change the default Qlik Replicate UI Server Service Log On user:

1. Open a command prompt and run the following commands:
 - a. Delete the existing user (if there is one), by running the following commands:

```
netsh http delete urlacl url=http://+:80/AttunityReplicate
netsh http delete urlacl url=https://+:443/AttunityReplicate
```
 - b. To add the user, run the following commands (where *username* is replaced by the name of the user):

```
netsh http add urlacl url=http://+:80/AttunityReplicate user=Username
netsh http add urlacl url=https://+:443/AttunityReplicate user=Username
```
2. Edit the **Qlik Replicate UI Server** service to log on with the new user.
3. Restart the **Qlik Replicate UI Server** service.

Windows software requirements

To install the Qlik Replicate Server and Console on a Windows computer, you must have the following installed on your system:

- .NET Framework 4.8 or later
- Visual C++ Redistributable for Visual Studio 2015. If it is not installed or if an older version is installed, it will be installed automatically during installation.
- TLS 1.2 or later must be supported in the underlying OS.



On Windows Server 2012 R2, TLS 1.2 should be turned on by default. If it is not, refer to the Microsoft online help for instructions on how to turn it on.

See also: *TLS Support (page 68)*

- For a list of supported browsers, see *Supported browsers (page 1327)*.
- Port 443 must be opened for outbound and inbound communication.

For a list of supported operating systems, see *Supported platforms (page 1313)*.

Installing or upgrading Qlik Replicate on Windows

Install Qlik Replicate using the `QlikReplicate_<version-build>_x64.exe` installation kit. This kit runs on Windows 64-bit (x64) environments. For a list of the Windows versions supported by Qlik Replicate, see *Windows software requirements (page 44)*.

Follow the instructions in the Setup wizard to install Qlik Replicate.

Later, if you need to start or stop the Qlik Replicate Server service, see the following section: *Starting and stopping the Qlik Replicate Server service on Windows (page 46)*



*In the setup wizard's **Replication Server Location** screen, one of the options is **Connect to a remote Linux Qlik Replicate Server**. You should only select this option if you have already installed Qlik Replicate Server on a Linux machine. If you select this option, you will be prompted for the IP address and port number of the Linux machine in the following screen.*

*For more information on installing Qlik Replicate Server on Linux, see *Setting up Qlik Replicate on Linux (page 49)*.*

*For information on the possible deployment configurations, see *Qlik Replicate UI server configurations (page 89)*.*

All of the data that is created when you use Qlik Replicate is stored in a directory called **data**. By default, this directory is located in the installation directory where you install Qlik Replicate. If you want to create the data directory in a different location, select this option in the installation wizard.



Qlik recommends using a block device (physical, virtual or iSCSI-based) for the **data** folder. Using NFS or SMB-based storage is not supported due to the associated latency which could greatly degrade the data transfer performance, as well as due to reduced reliability and compatibility issues.

If you elect to create the data directory in a different location, all command line actions must include the `-d your_data_directory_path` parameter where `your_data_directory_path` is the location of the "data" directory.

Upgrade instructions

To upgrade Replicate:

1. Choose one of the backup methods below.

Method 1:

- a. Stop all Replicate tasks and services.
- b. Back up the Replicate "data" directory to a location outside the product folder.

This method is recommended in most cases. However, if your "data" folder is very large and there is not enough space to back it up, then use **Method 2** below.

Method 2:

- a. Open a Windows command prompt and change the path to `<PRODUCT_DIR>\bin`.
- b. Then issue the following command:

```
repctl [-d data_directory] exportrepository
```

where `[-d data_directory]` is required when the Replicate "data" directory is in a non-default location.

2. Run the Qlik Replicate setup wizard.
3. If you notice an issue with the upgrade, you can either revert to the previous version as described below or do one of the following depending on which backup method you chose above.

If you chose Method 1:

- a. Stop all the Replicate services.
- b. Overwrite the "data" directory with the "data" directory that you backed up earlier.
- c. Start all the Replicate services.

If you chose Method 2:

Issue the following command:

```
repctl [-d data_directory] importrepository json_file=Full path to the exported *.json file
```

where `[-d data_directory]` is only required if you installed the Replicate "data" directory to a non-default location.

Example:

```
repctl -d f:/data importrepository json_file=C:\Temp\many_tables.json
```



The following server settings will not be affected:

- The **Store trace/verbose logging in memory, but if an error occurs write to the logs** logging option.
- Global and component-specific logging levels
- File Transfer Service settings

Reverting to a previous version

To revert to a previous version:

1. Back up the Replicate **data** directory to a location outside the product folder.
2. Uninstall the product and then reinstall to the same directory.
3. Once the installation is complete, stop all Replicate services.
4. Overwrite the **data** directory with the **data** directory that you backed up earlier.
5. Start all Replicate services.

Starting and stopping the Qlik Replicate Server service on Windows

In some cases you may need to stop and start the Qlik Replicate Server service. You must do this on the Windows computer where Qlik Replicate is installed.

To stop and start the Qlik Replicate Server service on Windows

- From the **Start** menu on the Windows computer where Qlik Replicate is installed, find **Qlik Replicate**; then select either **Stop Replicate Server** or **Start Replicate Server**.

Silently installing Qlik Replicate

Qlik Replicate can be installed silently (i.e. without requiring user interaction). This option is useful, for example, if you need to install Qlik Replicate on several machines throughout your organization.



Before commencing the installation, make sure that the [installation prerequisites](#) have been met.

The installation process consists of creating a response file and then running the silent installation.

Creating a response file

Before starting the installation, you need to create a response file.

To create the response file

1. From the directory containing the Qlik Replicate setup file, run the following command (note that this will also install Qlik Replicate):
`QlikReplicate_<version-build>_x64.exe /r /f1my_response_file`
where:
`my_response_file` is the full path to the generated response file.

Example:

```
QlikReplicate_<version-build>_x64.exe /r /f1C:\Replicate_install.iss
```

2. To change the default installation directory, open the response file in a text editor and edit the first **szDir** value as necessary.
3. To change the default data directory, edit the third **szDir** value as necessary.
4. Save the file as <name>.iss, e.g. *silent_inst_64.iss*.

Running the silent install

To silently install Qlik Replicate, open a command prompt and change the working directory to the directory containing the Qlik Replicate setup file. Then issue the following command (where <response file> is the path to the response file you created earlier):

Syntax:

```
<Replicate_setup_file> /s /f1my_response_file [/f2LOG_FILE]
```

Example:

```
C:\>QlikReplicate_<version-build>_x64.exe /s /f1C:\temp\1\Replicate_install.iss  
/f2C:\temp\1\silent_x64_install.log
```

If the installation was successful, the log file should contain the following rows:

```
[ResponseResult]  
ResultCode=0
```

Silently upgrading Qlik Replicate

Silently upgrading Qlik Replicate consists of creating a response file and then running the upgrade.

Creating a response file

Before starting the upgrade, you need to create a response file.

For instructions, see Step 1 of *Creating a response file* (page 46).

Running a silent upgrade

Before upgrading it is strongly recommended to back up the Replicate "Data" folder. To silently upgrade Qlik Replicate, open a command prompt and change the working directory to the directory containing the Qlik Replicate setup file.

Then issue the following command (where *my_response_file* is the path to the response file you created earlier):

Syntax:

```
<REPLICATE_SETUP_FILE> /s /f1my_response_file [/f2LOG_FILE]
```

Example:

```
C:\>QlikReplicate_<version-build>_x64.exe /s /f1C:\temp\1\Replicate_upgrade.iss  
/f2C:\temp\1\silent_x64_up.log
```

If the upgrade was successful, the log file should contain the following rows:

```
[ResponseResult]
ResultCode=0
```

Silently uninstalling Qlik Replicate

Silently uninstalling Qlik Replicate consists of creating a response file and then running the uninstall.

Creating a response file

Before starting the uninstall, you need to create a response file.

To create the response file

1. Copy the response file text below into a text editor.

Response file text:

```
[{9C614355-28A0-4C2A-98DF-DB9FD674826F}-DlgOrder]
Dlg0={9C614355-28A0-4C2A-98DF-DB9FD674826F}-SdWelcomeMaint-0
Count=3
Dlg1={9C614355-28A0-4C2A-98DF-DB9FD674826F}-MessageBox-0
Dlg2={9C614355-28A0-4C2A-98DF-DB9FD674826F}-SdFinish-0
[{9C614355-28A0-4C2A-98DF-DB9FD674826F}-SdWelcomeMaint-0]
Result=1
[{9C614355-28A0-4C2A-98DF-DB9FD674826F}-MessageBox-0]
Result=6
[{9C614355-28A0-4C2A-98DF-DB9FD674826F}-SdFinish-0]
Result=1
bOpt1=0
bOpt2=0
```

2. Save the file as *<name>.iss*, e.g. *silent_uninst_64.iss*.

Running a silent uninstall

To silently uninstall Qlik Replicate, open a command prompt and issue the following command (where `RESPONSE_FILE` is the path to the response file you created earlier and `LOG_FILE` is the path to the uninstall log file):

Syntax:

```
"C:\Program Files (x86)\InstallShield Installation Information\directory_containing_setup_file\setup.exe" /s /f1RESPONSE_FILE [/f2LOG_FILE]
```



The directory containing the Replicate setup file always ends with the following string:

```
DB9FD674826F
```

Example:

```
C:\>"C:\Program Files (x86)\InstallShield Installation Information\{9C614355-28A0-4C2A-98DF-DB9FD674826F}\setup.exe" /s /f1C:\temp\response.iss /f2C:\temp\1\silent_uninstall.log
```

If the uninstall was successful, the log file should contain the following rows:

```
[ResponseResult]
ResultCode=0
```

Changing the data directory location on Windows

This section explains how to change the location of the Qlik Replicate Data Directory. Such a procedure may need to be performed if the drive on which the current directory resides has insufficient space or if you are moving from a temporary POC setup to production, for example.

To change the location of the data directory

1. Stop the **Qlik Replicate UI Server** and **Qlik Replicate Server** services.
2. Move the data directory to a new location. For example:
`C:\Program Files\Attunity\Replicate\Data2`
3. Open the Registry and perform the following procedure:
 1. Browse to:
`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\AttunityReplicateConsole`
 2. Modify the **ImagePath** string as follows:
`"C:\Program Files\Attunity\Replicate\bin\RepUiCtl.exe" -d "C:\Program Files\Attunity\Replicate\Data2" service run`
 3. Browse to:
`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\QlikReplicateServer`
 4. Open the **ImagePath** string and add **-d <path_for_new_data_directory>** after the `repctl.exe` path. For example:
`"C:\Program Files\Attunity\Replicate\bin\repctl.exe" -d "C:\Program Files\Attunity\Replicate\Data2" service start name=Server address=127.0.0.1 port=3552`
4. Start the Qlik Replicate services.

3.3 Setting up Qlik Replicate on Linux

This section describes how to install Qlik Replicate on Linux and contains the following topics:

- [Replicate instances and services on Linux \(page 50\)](#)
- [Linux installation prerequisites and procedures \(page 51\)](#)
- [Replicate server procedures \(page 53\)](#)
- [Upgrading Qlik Replicate \(page 54\)](#)
- [Uninstalling Qlik Replicate \(page 56\)](#)
- [Working with additional Replicate instances \(page 57\)](#)
- [Managing Replicate instances \(page 59\)](#)

For information on supported Linux platforms, see [Supported Linux platforms \(page 1314\)](#).



- The login shell for the root user must be set to bash.
- Before running a `repctl` command in a Linux shell session for the first time, you need to set the environment, either by invoking the login script manually (using: `$ source arep_login.sh`) or by adding that line to the `~/.bashrc` script.
- For all command line operations involving files such as `arep_login.sh`, only the bash shell is supported.
- The commands for installing, upgrading and uninstalling Qlik Replicate must be run as root or using the `sudo` command.
- All of the commands and examples in this section assume that Qlik Replicate is being installed/upgraded/uninstalled as root.

Replicate instances and services on Linux

Replicate supports running multiple instances concurrently on the same Linux server with a single installation. This may be useful, for instance, if you need to run several tasks or groups of tasks, but with different Replicate server settings for each group or task.

Instances can be installed as Linux services, which means that the instance will be stopped in an orderly fashion when a server is shut down and restarted when a server is rebooted. Installing Replicate creates an initial instance, named `areplicate`, which is installed as a service.

Additional instances can be created after Replicate is installed. An additional instance will not be installed as a service if installed by a non-root user. A new instance will not be run when it is created, allowing for configuration first.

Replicate instances get their environment from several files:

Environment files

File	Description
<code>arep_login.sh</code>	Sets shell variables, e.g. <code>LD_LIBRARY_PATH</code> for all instances. This file should not be modified manually.
<code>site_arep_login.sh</code>	Created functionally null; can be modified manually to contain shell settings for all instances.
<code>instancename_arep_login.sh</code>	Created functionally null in the instance's data directory, for an instance-specific shell configuration.

Each instance has a unique name, uses its own ports, and has its own **data** directory. This means that when running commands such as `repctl`, you need to specify the instance-specific **data** directory in the command line so that the correct instance is affected.



- When starting and stopping an instance, it is strongly suggested to use the instance script rather than run `repctl` directly.
- All instances are run as the same user. Instances are listed in a file named `services_list.txt` in the Replicate root directory. This file should not be modified manually.
- When upgrading Replicate, all instances will be reinstalled and started, even if they were not running prior to the upgrade.

Linux installation prerequisites and procedures

The following topic lists the prerequisites for installing Qlik Replicate on Linux and explains how to perform the installation.

Prerequisites

- Run the following command to untar the Qlik Replicate RPM:
`tar xf areplicate_<version>_<buildnum>_Linux_X64.tar.gz`
- Copy the Qlik Replicate RPM file to any location on the Linux computer
- For the installation to succeed, at least one of the following must already exist in the system:
 - The user specified in the installation command
 - The `/home` dir
- Port 3552 (the default rest port) needs to be opened for outbound and inbound communication, unless you override it as described below.

Installing Qlik Replicate on Linux

The default installation directory for Qlik Replicate is: **`/opt/attunity/replicate`**

You can choose to install to a different directory as described below.

To install Qlik Replicate:

Run the following command:


```
[user=user] [group=group] [verbose=true|debug=true] [nocredentials=true]
[data=replicate_data_directory] [iport=number] [rport=number] rpm -i[vh] [--prefix
dirname] areplicate-<version-build>.x86_64.rpm
```

Example:

```
user=mike group=sales verbose=true iport=1024 rport=1025 rpm -i[vh] --prefix
/opt/mydir/ areplicate-6.4.0-102.x86_64.rpm
```

For description of the optional parameters, see the *Linux installation prerequisites and procedures (page 51)* table below.

Command parameters

Parameter	Description
[user=user]	<p>The default user under which Qlik Replicate is installed is attunity. You can choose to install the product under a different user by adding the prefix <code>user=user</code> to the command.</p> <p>See also the [nocredentials=true] parameter below.</p> <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;">  <i>The login shell of the Replicate user must be set to bash.</i> </div>
[pass=password]	<p>Sets the server password.</p> <p>Users will be prompted for this password when connecting to the Replicate server through the Replicate Web Console.</p> <p>You can either set this password when installing Replicate (recommended) or you can set it later as described in <i>Changing the server password (page 69)</i>.</p>
[group=group]	<p>The default group under which Qlik Replicate is installed is <code>attunity</code>. You can choose to install the product under a different group by adding the prefix <code>group=group</code> to the command.</p> <p>See also the [nocredentials=true] parameter below.</p>
[nocredentials=true]	<p>If the specified user or group is not defined locally (namely, in Active Directory), you must include this parameter in the command. Otherwise, the installation will fail.</p>
[data=replicate_data_directory]	<p>The default Qlik Replicate "data" directory is:</p> <p><code><product_dir>/data.</code></p> <p>Use this parameter to install the Replicate data directory to a non-default location.</p> <p>Example:</p> <p><code>data=/opt/mydata</code></p>
[--prefix dirname]	<p>Prefixes the <code>attunity/replicate</code> application directory with the path specified by <code>dirname</code>. So, for example, if <code>--prefix /opt/mydir</code>, then Replicate will be installed to:</p> <p><code>/opt/mydir/attunity/replicate</code></p> <p>This is only required when installing Qlik Replicate to a non-default path.</p>
[verbose=true debug=true]	<p>Specify <code>verbose=true</code> for additional information during the installation, or <code>debug=true</code> for detailed debug messages during the installation.</p>

Parameter	Description
[runservice=false]	<p>This parameter prevents the services from starting. The value can only be "false".</p> <p>This parameter can be used when installing/upgrading the product but not when creating a Replicate instance.</p>
[iport]	<p>The internal port, which must be a decimal number greater than 1023 but less than 65535.</p> <p>This parameter is required when creating a Replicate instance but optional when installing/upgrading the product.</p>
[rport]	<p>The REST port, which must be a decimal number greater than 1023 but less than 65535.</p> <p>This parameter is required when creating a Replicate instance but optional when installing/upgrading the product.</p>

The installation procedure will perform the following actions:

1. Create a new user and group named **attunity** (unless you chose to use a different user and a group and/or a user named **attunity** already exists).
2. Change the Qlik Replicate installation directory owner to the **attunity** user and group, or to your preferred user and group.
3. Install the application files.
4. Start the service (areplicate).



The Qlik Replicate/lib directory must precede the /usr/lib64 directory in the LD_LIBRARY_PATH environment variable.

The environment variable is set in the <product_dir>/bin/arep_login.sh file.

Example:

```
export LD_LIBRARY_PATH=/opt/attunity/replicate/lib:/usr/lib64
```

Additional changes to LD_LIBRARY_PATH in the files site_arep_login.sh and/or instancename_arep_login.sh should be made with caution.

Replicate server procedures

This section describes how to start and stop a Replicate instance, and how to determine whether an instance is running.

Verifying that a Replicate instance is running

To verify that a Replicate instance is running, run the following command:

```
/opt/attunity/replicate/bin/instancename status
```

Example:

```
/opt/attunity/replicate/bin/areplicate status
```

The output will be similar to the following:

```
running: /opt/attunity/replicate/bin/repctl -d /opt/attunity/replicate/data service start  
port=3550 rest_port=3552
```

Starting and stopping a Replicate instance

To start a Replicate instance:

Run the following command (shown using the default installation path):

```
/opt/attunity/replicate/bin/instancename start
```

Example:

```
/opt/attunity/replicate/bin/areplicate start
```

To stop an Replicate instance:

Run the following command (shown using the default installation path):

```
/opt/attunity/replicate/bin/instancename stop
```

Example:

```
/opt/attunity/replicate/bin/areplicate stop
```

Upgrading Qlik Replicate

This section explains how to upgrade an installation of Qlik Replicate. Before upgrading it is strongly recommended to back up the Replicate **data** directory (or directories when multiple instances are installed).

Upgrade instructions

1. Choose one of the backup methods below.

Method 1:

Stop all Replicate tasks [and instances](#).

Back up the Replicate "data" directory to a location outside the product folder. This method is recommended in most cases. However, if your "data" folder is very large and there is not enough space to back it up, then use **Method 2** below.

Method 2:

Issue the following command from the product **bin** directory:

```
repctl [-d data_directory] exportrepository
```

where *[-d data_directory]* is the path to the Replicate "data" directory when it's installed in a non-default location.

2. Run the following command:

```
[user=username] [group=groupname] [nocredentials=true] [verbose=true|debug=true] rpm -U  
[vh] [--prefix dirname] areplicate-<version-build>.x86_64.rpm
```

Where *[--prefix dirname]* is only required when Qlik Replicate is installed in a non-default directory.

For additional information on the available command parameters see the table: *Linux installation prerequisites and procedures (page 51)*.

3. If you notice an issue with the upgrade, you can either revert to the previous version as described below or do one of the following depending on which backup method you chose above.

If you chose Method 1:

1. Stop all Replicate instances.
2. Overwrite the "data" directory with the "data" directory that you backed up earlier.
3. Start all Replicate instances.

If you chose Method 2:

Issue the following command:

```
repctl [-d data-directory] importrepository json_file=Full path to the exported *.json file
```

where [-d *data-directory*] is the path to the Replicate "data" directory when it's installed in a non-default location.

Example:

```
repctl -d /opt/mydatadir/ importrepository json_file=/opt/mydatadir/many_tables.json
```

To revert to a previous version:

1. Back up the Replicate "data" directory to a location outside the product folder.
2. Uninstall the product and then reinstall to the same directory.
3. Once the installation is complete, stop all Replicate instances.
4. Overwrite the "data" directory with the "data" directory that you backed up earlier.
5. Start all Replicate instances.



When upgrading from a version prior to Replicate 6.1 where the **data** directory is in a non-default location, you need to add the following parameter to the upgrade command:

```
data=existing-data-directory
```

Example (when Replicate is installed in the default installation directory):

```
data=/opt/mydatadir/ rpm -U[vh] areplicate-6.4.0-102.x86_64.rpm
```



If the user specified in the original installation has since been deleted, the upgrade will fail.



As in the initial installation, if the new or existing user and/or group is not defined locally (namely, in Active Directory), you must include the `nocredentia1s=true` parameter in the command. Otherwise, the upgrade will fail.



When upgrading Replicate, all instances will be reinstalled and started, even if they were not running prior to the upgrade.

Resolving Configuration File Conflicts

During an upgrade, if the new version's **repctl.cfg** file contains different parameters than the existing file, and the existing file has been modified, you need to manually merge the new configuration file with the existing one.

In such a scenario the upgrade process will:

- Rename the file from the new version.
The file will be renamed **repctl.cfg.rpmnew** and installed in **<product_dir>/replicate/bin/** (the same directory as the **repctl.cfg** file).

- Issue the following warning (assumes the default installation path):

```
[root@b1dlinux-rh62 tmp]# rpm -Uvh areplicate-version-build.x86_64.rpm
```

```
Preparing... ##### [100%]
```

```
1:areplicate warning:
```

```
/opt/attunity/replicate/bin/repctl.cfg created as
```

```
/opt/attunity/replicate/bin/repctl.cfg.rpmnew
```

```
##### [100%]
```



When the configuration files need to be merged, you need to restart each instance manually after merging the files.

To complete the upgrade

1. Manually (and cautiously) merge the new parameters in the **repctl.cfg.rpmnew** file with the existing parameters in the **repctl.cfg** file. Save the **repctl.cfg** file.
2. Delete the **repctl.cfg.rpmnew** file by issuing the following command:

```
rm -f <product_dir>/bin/repctl.cfg.rpmnew
```
3. Restart each instance by running the following commands in the **<product_dir>/replicate/bin** directory:

```
./instancename start
```

Example:

```
./areplicate start
```

Uninstalling Qlik Replicate

To uninstall Qlik Replicate, run the following command:

```
[verbose=true|debug=true] rpm -e areplicate
```

To ensure that Qlik Replicate was removed from the computer, run the following command:

```
rpm -q areplicate
```

The output should be:

```
Package areplicate is not installed.
```



Uninstalling Replicate will not delete a Replicate instance's **data** directory, the `services_list.txt` file, or any other modified files.

Working with additional Replicate instances

Replicate supports running multiple instances concurrently on the same Linux server with a single installation.

For an overview of multiple instance support, see *Replicate instances and services on Linux* (page 50)

In this section:

- *Installing an additional instance of Replicate as a service* (page 57)
- *Uninstalling a Replicate instance installed as a service* (page 58)

Installing an additional instance of Replicate as a service

The installation directory of an instance depends on the user running the script.

- If the script is run as root:
 - The instance will be installed as a service
 - The installation directory of the service will be `/etc/init.d`
- If the script is run by a non-root user, the installation directory of the instance will be the replicate bin directory



Installing an instance will not start the instance.

Syntax:

```
[pass=server_password] data=data_directory iport=internal_port_number  
rport=rest_port_number arep.sh install instancename
```

Example:

```
pass=28dj20G43839&*! data=/opt/mydata iport=5342 rport=5341arep.sh install  
myreplicateinstance
```

Command parameters

Parameter	Description
[pass=password]	<p>Sets the server password.</p> <p>Users will be prompted for this password when connecting to the Replicate server through the Replicate Web Console.</p> <p>You can either set this password when installing Replicate (recommended) or you can set it later as described in <i>Changing the server password (page 69)</i>.</p>
data=replicate_data_directory	<p>Use this parameter to install the Replicate data directory to a non-default location.</p> <p>Example:</p> <pre>data=/opt/mydata</pre>
iport	The internal port, which must be a decimal number greater than 1023 but less than 65535.
rport	The REST port, which must be a decimal number greater than 1023 but less than 65535.
instancename	The name of the service to create.

Running the script will:

- Confirm not using instance name / port numbers / data directory that are already in use



Reinstalling an instance after it was just stopped and uninstalled may require a few minutes wait until tcp releases the port(s)

- Determine the data directory and port numbers, using the command line parameters or default values
- Create a configured copy of itself, named `instance`, installed in the instance installation directory
- Add an `instancename` record to a list of installed instances in **services_list.txt**
- Create a functionally empty instance specific file in the data directory named **instancename_arep_login.sh** for instance specific settings
- If run as root:
 - Create start/stop links for run levels as specified in the script (**chkconfig**)
 - Create a symbolic link to **/etc/init.d/instancename** in the Replicate bin directory (same name)
- If a server password is specified, the script will run `repctl` to set the admin UI server password

Uninstalling a Replicate instance installed as a service

To uninstall an instance of Replicate, run the following command:

```
instancename uninstall[=soft]
```

Running the script will:

- Stop the instance *instancename*
- If run as root:
 - Delete the start/stop links for *instancename* for run levels as specified in the script (**chkconfig**)
 - Delete *instancename* symbolic link from the replicate bin directory
- Delete *instancename* from the service installation directory
- Delete the *instancename* record from the list of installed instances



The script will not allow a non-root user to uninstall a service that was installed by root.

Managing Replicate instances

This section explains how to control (start/stop/restart) Replicate instances, check the running status of a Replicate instance, and change the location of a Qlik Replicate instance's **data** directory. Changing the **data** directory of an instance of Replicate might need to be performed if the drive on which the current directory resides has insufficient space or if you are moving from a temporary POC setup to production, for example.

Start/stop/restart a Replicate instance

To start, stop, or restart a Replicate instance, run the following command:

```
instancename start|stop|restart
```

Running the script will start/stop/restart Replicate running as *instancename*

Report running status of a Replicate instance

To find out the running status of a Replicate instance, run the following command:

```
instancename status
```

This will report whether *instancename* is running or not.

Change the data directory of a Replicate instance

To change the data directory of a Replicate instance, do the following:

1. Uninstall the instance.
2. Move the data directory to a new location.
3. Install an instance of replicate, specifying the data directory in the new location.

4 Security considerations

Qlik Replicate is tasked with replicating data within an organization, a task which involves reading from source endpoints (such as databases) and writing to target endpoints (including databases, files, and queuing systems).

This section provides important information for protecting the data that Qlik Replicate stores and replicates.

In this section:

- *Securing access to the Qlik Replicate Web UI (page 60)*
- *Setting up HSTS on Replicate (page 61)*
- *Setting Up Qlik Replicate console HTTPS support (page 63)*
- *Setting Up Qlik Replicate Server HTTPS support (page 64)*
- *TLS Support (page 68)*
- *Changing the server password (page 69)*
- *Protecting Replicate passwords (page 71)*
- *Client password considerations (page 74)*
- *Encrypting the user permissions file (page 74)*
- *Securing connections to endpoints (page 75)*
- *Log file encryption (page 75)*
- *Application security (page 77)*
- *Using external credentials (page 77)*
- *Controlling execution of user-defined commands (page 79)*

4.1 Securing access to the Qlik Replicate Web UI

Qlik Replicate offers the following Web UI configurations:

- A Windows-based Qlik Replicate UI Server service which offers granular user authorization based on a user's Active Directory identity and group membership. This service provides user interface functionality and communicates with the backend Qlik Replicate Server (on Windows or Linux). Connecting to this server is done using HTTPS with Windows Authentication.
- The Qlik Replicate Server on Windows or Linux can also serve the Web UI directly, but supports just a single user with a fixed role of administrator ('admin'). Connecting to this server is done using HTTPS with Basic Authentication.
See also *Configuration 3: Replicate UI Console and Replicate server running on Linux (page 90)*.

In line with current industry security standards, the Qlik Replicate web user interface enforces the use of HTTPS to protect against eavesdropping and data leakage. Using HTTPS requires a valid server certificate to be installed on the Qlik Replicate server machine.

4.2 Setting up HSTS on Replicate

HSTS is a web security policy mechanism that helps to protect websites against man-in-the-middle attacks such as protocol downgrade attacks and cookie hijacking. It allows web servers to declare that web browsers (or other complying Dilqam) should automatically interact with it using only HTTPS connections, which provide Transport Layer Security (TLS/SSL).

You can force the Replicate Web UI and/or the Replicate REST API connections to use HSTS (HTTP Strict Transport Security). To do this, run the commands described below.

All commands should be run from as Admin from the product **bin** folder.

Enabling HSTS on the Replicate UI Server

Command syntax

```
Repuictl.exe configuration set --static_http_headers header_list --rest_http_headers header_list
```

Parameters

Parameter	Description
--static_http_headers	The headers required to connect to the Replicate Web UI.
--rest_http_headers	The headers required to connect using the API.

Headers should be specified using the following format:

```
Repuictl.exe configuration set --static_http_headers "header1:value1" "header2:value2" --rest_http_headers "header1:value1" "header2:value2"
```

Example

```
Repuictl.exe configuration set --static_http_headers "Strict-Transport-Security:max-age=31536000; includeSubDomains;" --rest_http_headers "Strict-Transport-Security":"max-age=31536000; includeSubDomains;"
```

Disabling HSTS on the Replicate UI Server

You can also revert to regular HTTPS connections.

Command syntax

```
Repuictl.exe configuration set --static_http_headers ""|--rest_http_headers ""
```

Parameters

Parameter	Description
--static_http_headers	Use this parameter to revert the headers required to connect to the Replicate Web UI.
--rest_http_headers	Use this parameter to revert the headers required to connect using the API.

Example

Disable static_http_headers

```
Repuictl.exe configuration set --static_http_headers ""
```

Disable rest_http_headers

```
Repuictl.exe configuration set --rest_http_headers ""
```

Configuring HSTS on the Replicate C UI Server

When connecting directly to the Replicate C UI Server (i.e. when Replicate Server is running on Linux), you need to add the **rest_handler_response_additional_headers** and the **static_resources_response_additional_headers** to the **repctl.cfg** file. The default location of this file is `/opt/attunity/replicate/bin/` on Linux (and `C:\Program Files\Attunity\Replicate\bin` on Windows). If you wish to remove these response headers at a later date, simply delete them.

See also: *Qlik Replicate UI server configurations (page 89)*.



Any changes to the **repctl.cfg** file (such as editing/removing the headers) require you to restart the Replicate services once you have made your changes.

Example:

The following example shows the **repctl.cfg** file with both the **rest_handler_response_additional_headers** and the **static_resources_response_additional_headers**. Each of these headers is defined with X-XSS-Protection, X-Frame-Options, and Strict-Transport-Security response headers.

- The **X-XSS-Protection** response header can be used to block reflected XSS attacks.
- The **X-Frame-Options** response header defends against "clickjacking" by instructing the browser to prevent any site with this header in the response from being rendered within a frame (a potential security issue).
- The **Strict-Transport-Security** response header instructs the browser to treat the domain as an HSTS host for a year (there are approximately 31536000 seconds in a year). The optional `includesubDomains` directive means that subdomains (i.e. `secure.myhealthcare.example.com`) should also be treated as an HSTS domain.

```
{  
  "port": 3552,  
  "plugins_load_list": "repui",  
  "mandatory_plugins": "repui",  
  "logger_status": {  
    "REST_SERVER": "DEFAULT",  
    "REPUI_PLUGIN": "DEFAULT"  
  },  
}
```

```
"rest_handler_response_additional_headers": "X-XSS-Protection: 1;
mode=block\r\nX-Content-Type-Options: nosniff\r\nX-Frame-Options:
SAMEORIGIN\r\nStrict-Transport-Security:max-age=31536000;
includeSubDomains;\r\n",
"static_resources_response_additional_headers": "X-XSS-Protection: 1;
mode=block\r\nX-Content-Type-Options: nosniff\r\nX-Frame-Options:
SAMEORIGIN\r\nStrict-Transport-Security:max-age=31536000;
includeSubDomains;\r\n",
"verbose":false,
"session_timeout_seconds":60,
"check_for_expired_session_interval_seconds":50,
"login_pam_libpam_full_path":"","
"login_pam_service_name": "system-auth",
"enable_execute_user_defined_commands": false
}
```

4.3 Setting Up Qlik Replicate console HTTPS support

Industry-standard security practices dictate that web user interface for enterprise products must use secure HTTP (HTTPS). Qlik Replicate enforces the use of HTTPS and will not work if HTTPS is configured incorrectly.

As Qlik Replicate uses the built-in HTTPS support in Windows, it relies on the proper setup of the Windows machine it runs on to offer HTTPS access. In most organizations, the IT security group is responsible for generating and installing the SSL server certificates required to offer HTTPS. It is strongly recommended that the machine on which Replicate is installed already has a valid SSL server certificate installed and bound to the default HTTPS port (443).

Checking if an SSL certificate is installed

To check whether an SSL certificate is installed, you can use the following command:

```
netsh http show sslcert | findstr /c:":443 "
```

If an SSL certificate is installed, the output should look like this:

```
netsh http show sslcert | findstr /c:":443 "
IP:port : 192.168.1.13:443
IP:port : 192.168.1.11:443
IP:port : [fe80::285d:599c:4a55:1092%11]:443
IP:port : [fe80::3d0e:fb1c:f6c3:bc52%23]:443
```

With a valid SSL certificate installed, the Qlik Replicate web user interface will automatically be available for secure access from a web browser using the following URL:

```
https://<machine-name>/attunityreplicate
```

Using the self-signed certificate

Due to the way the HTTPS protocol works, there is no way for Qlik Replicate to automatically provide and install a valid SSL server certificate. Still, in the event that no SSL server certificate is installed, Qlik Replicate automatically generates and installs a self-signed SSL server certificate (as a temporary measure). This certificate is generated on the Replicate machine and cannot be exported or used elsewhere.

It should be noted that browsers do not consider the certificate to be valid because it was not signed by a trusted certificate authority (CA).

When connecting with a browser to a server that uses a self-signed certificate, a warning page is shown informing you that the connection is not secure or similar (depending on the browser).

The warning page informs you that the certificate was signed by an unknown certificate authority. All browsers display a similar page when presented with a self-signed certificate. If you know that the self-signed certificate is from a trusted organization, then you can instruct the browser to trust the certificate and allow the connection. Instructions on how to trust the certificate vary between browsers and even between different versions of the same browser. If necessary, refer to the help for your specific browser.



Some corporate security policies prohibit the use of self-signed certificates. In such cases, it is incumbent upon the IT Security department to provide and install the appropriate SSL server certificate (as is the practice with other Windows products such as IIS and SharePoint). If a self-signed certificate was installed and needs to be removed, then the following command can be used:

```
$ netsh http delete sslcert ipport=192.168.1.13:443
```

where ipport should be replaced with the ip:port combination generated by the netsh command shown in [Checking if an SSL certificate is installed](#) (page 63).

4.4 Setting Up Qlik Replicate Server HTTPS support

The Qlik Replicate Server which runs on both Windows and Linux uses the OpenSSL HTTPS implementation. The Qlik Replicate Server automatically generates a self-signed certificate server but it allows you to replace it with a server certificate signed by a trusted certificate authority. This section describes how to accomplish this on Linux and Windows.

Replacing the self-signed SSL certificates on Linux

When Qlik Replicate Server starts for the first time, it checks the `<product-dir>/ssl/data` directory for the presence of certificates. If the `ssl` folder is not found, it will then check the `<product-dir>/<data-directory>/ssl/data` directory (or directories when [running multiple Linux instances](#)) for the certificates.

If there are no certificates, it will create the following self-signed certificates:

- **agent-ca.pem** - The CA certificate
- **agent-certificate.pem** - The public certificate
- **agent-private-key.pem** - The private key data
- **agent-private-key-passphrase.dat** - The private key passphrase



When working with multiple instances, instead of creating a separate set of certificates for each instance in `<product-dir>/<data-directory>/ssl/data`, you can create a single set of certificates in `<product-dir>/ssl/data`. This way, instead of managing multiple sets of certificates for each instance, you only need to create/manage a single set of certificates.

You can replace the default self-signed certificates with your own, as follows:

1. Stop the Qlik Replicate Server service.
2. Create the required certificates using names that are identical to the certificates listed above.
3. Copy the certificates to the **ssl/data** directory (`<product-dir>/<data-directory>/ssl/data` by default).
4. Edit the **agent-private-key-passphrase.dat** file as follows:
`/c\ear:PRIVATE_KEY_PASSWORD`

Example:

```
/c\ear:12345
```

When Qlik Replicate Server starts it will scramble the private key passphrase as shown in *Examples of the Scrambled Private Key Password* (page 65).

5. Start the Qlik Replicate Server service.

For information on stopping and starting Qlik Replicate Server, see *Setting up Qlik Replicate on Windows* (page 40) and *Setting up Qlik Replicate on Linux* (page 49).

Examples of the Scrambled Private Key Password

The scrambled private key passphrase stored in the `agent-private-key-passphrase.dat` file will look similar to this:

```
{S:DEA326D0DF190430975DE44CFBD6FDFD21883C10E7651081B3B5A0A7404BB97DB520876F60390B51300C831C82DE871CF8BA22393D8DD9B359DD5A93C5956710AD2546E188155482452235C5D91B430D151E3DDA7381CA3E}
```

Replacing the self-signed certificate on Windows

The instructions below are intended for organizations who wish to replace the self-signed certificate generated by the Replicate UI Server on Windows with their own certificate. This is achieved by removing the self-signed certificate and then importing the new certificate.

See also *Setting Up Qlik Replicate console HTTPS support* (page 63).

Before starting, make sure that the following prerequisites have been met:

- The replacement certificate must be a correctly configured SSL PFX file containing both the private key and the certificate.
- The common name field in the certificate must match the name browsers will use to access the machine.

To remove the self-signed certificate created by Qlik Replicate:

1. Stop the Qlik Replicate Server and Qlik Replicate UI Server services.
2. Open a command prompt (using the "Run as administrator" option) and change the path to the Replicate *bin* directory. The default path is:
`C:\Program Files\Qlik\Replicate\bin.`
3. Run the following command:
`RepUiCtl.exe certificate clean`

To create and import your own certificate, open a command prompt and do the following:

1. Create the private key file:
`openssl genpkey -algorithm RSA -out private.key -outform PEM`
2. Create the **.crt** file based on the private key file created earlier:
`openssl req -new -addext "subjectAltName = DNS:SSL-TEST.exampledomain.com" -key private.key -x509 -days 365 -out cert.crt`
3. Create the **.pfx** file based on the above:
`openssl pkcs12 -export -certpbe PBE-SHA1-3DES -keypbe PBE-SHA1-3DES -nomac -inkey private.key -in cert.crt -out cert.pfx`



When prompted for the Organizational Unit Name and the Common Name, make sure to specify the same DNS that was used to create the .crt. To view the DNS, run the following command:

nslookup %computername%

4. Import the .pfx certificate into the **My** certificate store (Personal certificates):
`netsh http del sslcert ipport=0.0.0.0:443`
`netsh http add sslcert ipport=0.0.0.0:443 certhash=<certificate thumbprint>appid={4dc3e181-e14b-4a21-b022-59fc669b0914}`

4.5 FIPS compliance



Supported from Replicate May 2022 Service Release 02 only.

FIPS (Federal Information Processing Standards) is a set of standards developed by the United States Federal Government for use in computer systems. FIPS 140-2 is the subset of standards which defines approved encryption algorithms used for handling sensitive information. Replicate will work in FIPS mode if the following prerequisites have been met:

- A FIPS-compliant Replicate version must be installed. To obtain the FIPS-compliant version, contact [Qlik Support](#).



Upgrading from a non FIPS-compliant Replicate is not supported. You need to uninstall your current version before installing the FIPS-compliant version. Once you have installed the FIPS-compliant version, the next Replicate version that you upgrade to does not need to be FIPS-compliant (as you will already have all the FIPS-compliant software on your machine).

- The operating system on which Replicate is installed is running in FIPS mode.

FIPS-compliant endpoints

FIPS compliance in Replicate endpoints depends on whether the database or ODBC vendor supports the FIPS cryptographic standard. As this is not always the case, only the endpoints listed in the table below are FIPS compliant.

FIPS compliant endpoints

Endpoint	Comments
File source and target	-
IBM DB2 for z/OS source	<p>The SSL connection is FIPS compliant. To connect using SSL, set the following internal parameters in the endpoint's Advanced tab:</p> <ul style="list-style-type: none"> • <code>UseSSL</code> Enables SSL • <code>SSLClientKeystoredb</code> The SSL keystore database file that is used for an SSL connection with or without the CERTIFICATE authentication. • <code>SSLClientKeystash</code> The fully qualified path of the stash file (<code>.sth</code>), which stores an encrypted password to the key database file. <p style="text-align: center;">-OR-</p> <p><code>SSLClientKeystoreDbPassword</code> The password of the keystore in string format. This password is defined when the keystore is generated.</p> <p><i>See also: Setting advanced connection properties (page 285)</i></p>
IBM DB2 for iSeries source	<p>Supported with Replicate on Windows only. The SSL connection is FIPS compliant. To connect using SSL, set the <code>useSSL</code> internal parameter in the endpoint's Advanced tab.</p> <p><i>See also: Setting advanced connection properties (page 262)</i></p>

Endpoint	Comments
IBM DB2 for LUW source	<p>The SSL connection is FIPS compliant. To connect using SSL, set the following internal parameters in the endpoint's Advanced tab:</p> <ul style="list-style-type: none"> • <code>UseSSL</code> Enables SSL • <code>SSLClientKeystoreDb</code> The SSL keystore database file that is used for an SSL connection with or without the CERTIFICATE authentication. • <code>SSLClientKeystash</code> The fully qualified path of the stash file (<code>.sth</code>), which stores an encrypted password to the key database file. <p>-OR-</p> <p><code>SSLClientKeystoreDbPassword</code> The password of the keystore in string format. This password is defined when the keystore is generated.</p> <p>See also: <i>Setting advanced connection properties (page 273)</i></p>
Microsoft Azure ADLS target	<p>Supported when using Azure Data Lake Storage (ADLS) Gen1 only.</p> <p>See also: <i>Setting general connection properties (page 871)</i></p>
Microsoft Azure Event Hubs target	-
Oracle source and target	<p>For instructions on how to enable FIPS, see https://docs.oracle.com/cd/E28271_01/network.1111/e10746/asoappe.htm</p>
Sybase ASE source	<p>To enable FIPS, set the <code>EnableFIPS</code> internal parameter in the endpoint's Advanced tab to <code>true</code>.</p>
Vertica target	<p>Supported with Replicate on Linux only. Requires a FIPS compliant driver, which can be downloaded from: https://www.vertica.com/download/vertica/client-drivers/</p>

4.6 TLS Support

Replicate supports TLS 1.2 or later. However, support for TLS 1.3 is dependent on the Replicate server you choose to work with.

The native Replicate server listener (port 3552) supports TLS 1.2 or later. However, if you work against the Replicate UI server (which uses the OS HTTP stack), you will need to make sure the OS HTTP includes support for TLS 1.2 and later. Information pertaining to this should be found on the OS vendor site.

For more information on Replicate server configurations, see *Qlik Replicate UI server configurations (page 89)*.

4.7 Changing the server password

This topic explains how to change the Qlik Replicate Server password using the Replicate CLI. Note that all commands described in this section should be run as administrator from the product *bin* directory.

The Qlik Replicate Server has a fixed 'admin' user with an automatically generated random password that is stored in the *mk.dat* file. The password is unknown, unique and safe. The **Qlik Replicate UI Server** service always connects to the **Qlik Replicate Server** service using the 'admin' user. When both services run on the same machine the admin password is accessible to both servers, so there is no need to specify this password explicitly.

When Qlik Replicate Server runs on a different machine or when a remote Qlik Replicate client needs to communicate directly with a remote Qlik Replicate Server, the server password must be known to both sides.

The server password must be strong enough to prevent brute-force or dictionary attacks. As such, it must satisfy the following criteria:

Must contain at least:

- 16 characters
- One uppercase letter [A-Z]
- One lowercase letter [a-z]
- One digit [0-9]

Cannot contain:

- Special keyboard characters (e.g. !@#)\$)
- Non-Latin letters
- Spaces

You can generate a strong random password using the *genpassword* CLI utility described below, using a third-party utility, or even manually; in fact, it's not important how the password is generated, as long it meets the aforementioned requirements.



- Passwords set with previous versions will continue to work, as the validity check is only performed when setting a new server password.
- Scripts that automate the setting up of Qlik Replicate servers (such as when setting up docker images) should be reviewed and edited if necessary to ensure password validity.
- When running multiple Replicate Linux instances, this procedure needs to be repeated for each instance (as each instance has its own data directory).
For information on installing multiple Replicate Linux instances, see *Replicate instances and services on Linux* (page 50)

To generate a strong random password:

Run the following command:

```
repctl genpassword
```

A 16 character password that satisfies the strong-password criteria will be generated.

To change the server password using a script:

1. Run the following command:

```
repctl [-d data-directory] SETSERVERPASSWORD new_password
```

where *data-directory* is the name of the Replicate data directory.
The default path is `<product_dir>/data`.



You only need to include `-d data-directory` in the command if you changed the default data directory (e.g. when installing multiple Replicate Linux instances).

2. Restart the Qlik Replicate services (Windows) or the Qlik Replicate instance (Linux).

To change the server password interactively:

1. Run the following command:

```
repctl [-d data-directory]
```
2. Press [Enter] and then type the following:

```
SETSERVERPASSWORD new_password
```
3. Press [Enter] again to set the password.
4. Restart the Qlik Replicate services (Windows) or the Qlik Replicate instance (Linux).



Notes:

When the Qlik Replicate .NET UI Server is running on one machine and the Qlik Replicate Server is running on another, the Qlik Replicate Server password must be the same on both machines. The password is used during the SSL handshake to establish a secure connection between the participating machines.

4.8 Protecting Replicate passwords

Replicate stores secrets (e.g. passwords and keys) in its internal repository, enabling it to perform secure operations during runtime, such as connecting to an endpoint, connecting to a web server, connecting to an email server, connecting to a remote file transfer server, and so on.

As a rule, all UI values displayed as asterisks are stored encrypted and never transmitted or exposed by the API. For instance, when exporting task definitions, all passwords are encrypted and can only be decrypted if the source machine's `mk.dat` file - and possibly also the account or machine identity (on Windows) - is copied to the target machine. See also *Master key considerations when exporting and importing tasks (page 73)*.

Secrets that appear in Replicate settings are stored in an encrypted form with the following properties:

- The secret is encrypted using the AES-256 encryption algorithm.
- The encryption key, also known as the 'master key', is a 256-bit key that is stored (and protected as described later) in the master key file (`mk.dat`).
- The encryption process uses a salt that depends on the name and type (or 'context') of the protected entity. This prevents reuse attacks.
- The encryption process uses a nonce so that the same secret is always encrypted differently. This prevents dictionary attacks.

When exported to a JSON file, secrets appear in the following format:

```
"{Zxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx}"
```

Where:

- 'Z' is a fixed character
- 'x' a protection method indicator
- 'xxxxxxxx...' is a hexadecimal representation of the encrypted secret

Upon import, if a secret is provided in clear text (i.e. not in the format shown above), it is automatically encrypted and stored in the protected format.

The master key used in encrypting secrets is stored in the master key file (`mk.dat`) described below.

The master key file

The master key file is a binary file that stores the root secrets of Replicate. These root secrets currently include:

- The AES 256-bit master key, which is used for encrypting passwords and other secrets in Replicate settings.
- The server admin password, which is used (by the UI server) to access the replication server remotely.



*When running multiple Replicate Linux instances, you need to enter the server admin password for each instance. For details, see *Installing an additional instance of Replicate as a service (page 57)*.*

- The default location for the master key file is the product data folder - by default `<product-dir>/data` - or data directories when [multiple instances of Replicate](#) are installed on Linux. If the server admin password does not exist, it is automatically created with randomly generated passwords that are safe but unknown. The user can change the server admin password as well as the master key to known values, if needed (e.g to connect to the replication server remotely).
For more information, see *Changing the server password (page 69)*.

When Replicate is set to run on a cluster using the same settings and storage, the `mk.dat` file should also be the same (or shared). Similarly, if the Replicate settings are exported in order to be moved to a different machine, the `mk.dat` file should also be moved in order to avoid the need to reenter all secrets.

The procedure for changing the master key as well as measures that can be taken to protect the file containing the master key are described in **Changing and protecting the master key**.

Changing and protecting the master key

This section describes how to change the master key as well as how to prevent unauthorized access to the master key file.

Changing the master key replacement

The master key is originally a randomly generated 256-bit key. It can be changed as described below immediately after installation with no consequences. However, if you change the master key after endpoints or tasks are already configured, all stored secrets will no longer be valid (as they were encrypted using the old key). Therefore, after changing the master key, you need to reenter the passwords in all the relevant places.



When running multiple Replicate Linux instances, this procedure needs to be repeated for each instance (as each instance has its own data directory).

For information on multiple Replicate Linux instances, see [Replicate instances and services on Linux \(page 50\)](#).

To change the Master Key:

1. Stop any running tasks.
2. Stop the Replicate services or [instance\(s\)](#) on Linux.
3. Open a command prompt as an administrator/root.
4. Change the working directory to the product "bin" directory and then run the following command (on Linux run `./repctl ...`):
`repctl [-d data-directory] setmasterkey your_new_master_key [master_key_scope=scope]`
where `data-directory` is the name of the Replicate data directory (by default: `data`).



You only need to include `-d data-directory` in the command if you changed the default data directory (e.g. when installing multiple Replicate Linux instances).



In order not to have the key appear in the shell history, you can use the command interface (on Linux run `./repctl`):

```
repctl {enter}
setmasterkey your_new_master_key [master_key_scope=scope] {enter}
quit {enter}
```

See *Protecting the master key file from misuse* (page 73) for the `master_key_scope` options.

Example:

```
repctl -d opt/mydatadir setmasterkey 78543043vuiyfyrf64454555jy65 master_key_scope=1
```

5. Start the Qlik Replicate Server or [instance\(s\)](#) on Linux.
6. Reenter the access passwords in all endpoints.
7. Start the tasks.

Protecting the master key file from misuse

Access to the `mk.dat` file is restricted to administrators (Windows) or to the users in the group under which the product was installed (Linux). Care should be taken to maintain this file protection.

The `mk.dat` file is always encrypted by some built-in, fixed key. On Windows, there are two additional options for preventing unauthorized access and use of the `mk.dat` file. These are as follows:

- Tying the `mk.dat` to the Machine Profile - With this option, the `mk.dat` file can only be used on the Windows machine where it was created. Using the file from a different machine will generate an error. This option is not appropriate for clusters that share the `mk.dat` file as only one machine will be able to read it.
- Tying the `mk.dat` to a User Profile - With this option, the `mk.dat` file can only be used under the identity that was used while creating it (typically the user that installed the product). For this option to work, the Replicate services must be modified to run under the same user. If the user has a roaming profile, then the `mk.dat` file can be used on multiple machines (e.g. in a cluster).

These options are specified in the `master_key_scope` option of the `setmasterkey` command. They should be set at the same time that the master key is set since any such change invalidates the stored passwords.

The master key scopes are:

- 1 (Constant) - The default. The `mk.dat` file can be used wherever it is copied to.
- 2 (User) - The `mk.dat` file can only be used under the same account as the one that was used when creating it.
- 3 (Machine) - The `mk.dat` file can only be used on the same machine where it was created.

Master key considerations when exporting and importing tasks

To be able to export tasks from one machine and then import them to another, the same master key must exist on both machines. This means that if you change the master key on one machine, you must also change it on the other machine.



Replicate enforces strict access restrictions to the mk.dat file. Consequently, in order to export a task, you will also need to open the command prompt as an administrator (on Windows) or the product account (Linux).

For more information on importing and exporting Replicate tasks, see *Exporting tasks* (page 126).

4.9 Client password considerations

Qlik Replicate follows industry best practices for securing credentials, including the defense-in-depth practice of keeping all credentials encrypted in memory except for the short time they need to be used.

Still, third-party database clients may not adhere to these best practices and leave clear passwords in process memory. Qlik cannot take responsibility for such cases and advises taking this concern up with the database client vendor in the event that such a vulnerability is discovered.

4.10 Encrypting the user permissions file

User permissions are stored in the following repository file:

```
<product_dir>\Data\GlobalRepo.sqlite
```

To prevent unauthorized access of this file, you can encrypt it using the procedure described below. After you perform the procedure, the repository file will be encrypted with the AES-256 bit cipher.



The length of any passwords specified during the procedure must be at least 32 characters.

To encrypt the repository file:

1. Open a command prompt as administrator and change the working directory to:

```
<product_dir>\bin
```
2. Run the following command to set the master user key:

```
repuictl.exe masterukey set --password your_MasterUserPassword
```

Example:

```
repuictl.exe masterukey set --password ANqagYERP3UKmGLK6UNuMqrkAGxwH8FM
```
3. Restart the Qlik Replicate Server service.
4. Run the following command to set the repository password:

```
repuictl repository setpassword --master-user-password your_MasterUserPassword --repository-password your_RepositoryPassword
```

Example:

```
repuictl repository setpassword --master-user-password ANqagYERP3UKmGLK6UNuMqrkAGxwH8FM --repository-password 12345678901234567890123456789000
```



Steps 1-4 only need to be performed the first time you want to encrypt the repository file. If you subsequently need to decrypt the repository file and then re-encrypt it, they are not required.

5. Run the following command to encrypt the repository:
`repuictl.exe repository secure --on --master-user-password your_MasterUserPassword`
Example:
`repuictl.exe repository secure --on --master-user-password
ANqaGYERP3UKmGLK6UNuMqrkAGxwH8FM`
6. Restart the Qlik Replicate Server service.

To disable encryption for the repository:

- Run the following command:
`repuictl repository secure --off --master-user-password your_MasterUserPassword`

For information on setting user permission, see *User permissions* (page 1274).

4.11 Securing connections to endpoints

Qlik Replicate communicates with the source and target endpoints (typically databases) using either the vendor provided client package or via a standard ODBC driver.

Qlik does not implement the network protocol (see important exceptions below) and for this reason, Qlik Replicate generally relies on the vendor of the source or target endpoint to offer encryption. When setting up endpoint connections, the user is able to specify any connection properties required for the use of encryption; these properties are, invariably, vendor-specific properties. In some cases, use of encryption requires system-level settings (such as adding a certificate to the machine trust store) which are outside of the scope of Qlik Replicate. Users are referred to the operation manual of the source or target endpoint for details on how to set up encrypted client communication with the server.

One exception to the previous paragraph is for endpoints based on the Qlik Connect product (endpoints on zOS, iSeries, HP NonStop and HP OpenVMS). In this case, the network encryption is implemented by the Qlik Connect product and is based on the AES encryption algorithm with a 256-bit key.

Another exception is for endpoints that work over HTTP. In these cases, the user is advised to ensure that the endpoint server is configured to offer HTTPS and then use the appropriate `https://` based URL when setting up the connection.

4.12 Log file encryption

When the logging level is set to Trace or Verbose, sections in a Replicate log file that might contain customer data will be encrypted while the rest of the log will remain in clear text.



Setting the logging level to Verbose for a prolonged period will consume a large amount of disk space and should therefore be avoided unless explicitly requested by Qlik Support.

In some support cases, R&D may require the ability to decrypt the log in order to analyze the problem. In such cases, Qlik Support will ask you to provide the task or environment-level encryption key. The encryption key is automatically generated and kept in a **log.key** file in the following locations:

- For decrypting a task's log file: <REPLICATE_INSTALL_DIR>\data\tasks\<task-name>
- For decrypting non-task related log files (e.g. server logs): <REPLICATE_INSTALL_DIR>\data

Should you wish to review the logs before sending to Qlik Support, you can decrypt them via the Replicate CLI as described below.

Decrypting Task Logs:

1. Open the Replicate command line from Start menu (**Qlik Replicate > Qlik Replicate Command Line**) or open a command prompt “as administrator” and switch to the following directory:

```
<REPLICATE_INSTALL_DIR>\bin>
```

2. Run the following command:

Syntax:

```
repctl dumplog <log-file-path> <log.key-path> [> <path-to-output-file>]
```

Example:

```
repctl dumplog "C:\Program Files\Attunity\Replicate\data\logs\tasks\MyTask.log"  
"C:\Program Files\Attunity\Replicate\data\tasks\MyTask\log.key" > "C:\Program  
Files\Attunity\Replicate\data\decrypted.log"
```

Decrypting Non-Task Logs:

1. Open the Qlik Replicate command line from **Start** menu (**Qlik Replicate > Qlik Replicate Command Line**) or open a command prompt “as administrator” and switch to the following directory:

```
<REPLICATE_INSTALL_DIR>\bin>
```

2. Run the following command:

Syntax:

```
repctl dumplog <log-file-path> <log.key-path> [> <path-to-output-file>]
```

Example:

```
repctl dumplog "C:\Program Files\Attunity\Replicate\data\logs\repsrv.log" "C:\Program  
Files\Attunity\Replicate\data\log.key" > "C:\Program  
Files\Attunity\Replicate\data\decrypted.log"
```



The encryption key file (**log.key** file) can be used to decrypt any log that was encrypted with it, not just the log for which Qlik Support requested the encryption key file. If this is a concern, you can generate a **log.key** file that will only be valid for the specific log requested by Qlik Support.

To do this:

1. Delete the **log.key** file.
2. Restart the task or Replicate Server service (depending on the log type required) to generate a new **log.key** file.
3. Send the requested log file and the **log.key** file to Qlik Support.
4. Delete the **log.key** file.
5. Repeat steps 2-4 if you need to provide additional encrypted logs.

4.13 Application security

As a leading provider of enterprise-class big data management solutions, Qlik understands that application security is of paramount importance. With the integration of Static Code Analysis into the development lifecycle, the code is rigorously tested for vulnerabilities before each product release.

4.14 Using external credentials

Many organizations prefer to keep secrets in a dedicated "vault" as a means of protecting against unauthorized privileged account access, impersonation, fraud, and theft. Storing secrets in a vault also eliminates manually intensive, time consuming and error prone administrative processes.

Replicate can be easily configured to interface with such vaults, thereby adding an additional layer of security while easing administration of endpoints. In Replicate, secret fields are any fields where the input is masked with asterisks. Such fields include database user passwords, user names, Amazon S3 secret keys, and so on.

It is also recommended to use the Replicate Add-ons API which allows you to take advantage of Replicate's memory management and logging capabilities, while eliminating the need to create independent memory and logging routines.

For more information, see *Replicate add-ons API (page 1334)*.

To facilitate integration with an external vault, Qlik provides the following files:

```
<INSTALL_DIR>\addons\samples\MyPasswordProvider.c
```

```
<INSTALL_DIR>\addons\include\ar_addon_password_provider.h
```



The path is the same on Linux, but with slashes (/) instead of backslashes (\). Similarly, although the explanation below refers to a DLL file (Windows), on Linux this should be an SO file.

Limitations and considerations

- Secrets starting with "lookup::" are not supported
- Secret length cannot exceed 4 KB
- Loading more than one passwordProvider will result in a fatal error

Supported endpoints

Endpoints that support both user name and password replacement

The following endpoints support both user name and password replacement:

- Amazon RDS for SQL Server source
- Microsoft Azure SQL (MS-CDC) source



Only when using SQL authentication or Azure Active Directory authentication

- Microsoft Azure SQL Managed Instance source



Only when using SQL authentication or Azure Active Directory authentication

- Microsoft SQL Server source
- Microsoft SQL Server (MS-CDC) source
- Oracle source, except the following:
 - Secret store encryption entries
 - ASM parameters (if the redo logs are stored in ASM)

Endpoints that support password replacement

All endpoints support password replacement, with the following exceptions:

- File source
- File target
- Hadoop source, but only when the **Authentication** type is Kerberos.
- Microsoft SQL Server source, but not for the **Replicate has file-level access to the backup log files** option.
- ODBC source, but not if the password is specified in the **Connection string**.
- ODBC with CDC, but not if the password is specified in the **Connection string**.
- SAP Application (DB) source, but only in the fields supported by the backend endpoint.
- Log Stream target

Changing the default addon name

You can change the default addon name (*MyPasswordProvider*) simply by renaming the `<INSTALL_DIR>\addons\samples\MyPasswordProvider\MyPasswordProvider.dll` file as desired. Note that if you rename the DLL, you may also need to specify the new name when editing the `addons_def.json.sample` file described [in Step 3 below](#) (depending on which parameters you use).

Step 1: Edit the MyPasswordProvider.c file

Edit the file `MyPasswordProvider.c` as follows:

1. Locate the following line:

```
static AR_ADDONS_STATUS get_secret(char *name, const char *lookup_key, char *secret, int secret_len, char *principal, int principal_len)
```

For a description of these fields, refer to the `ar_addon_password_provider.h` file.
2. Replace the code below this line (starting `AR_AO_LOG` and ending `AR_ADDONS_STATUS_SUCCESS;`) with the code required to access your Secrets Vault.

Step 2: Compile the code

Compile the `MyPasswordProvider.c` file into a DLL/SO. Note that on Windows the file should be compiled in "Release x64" mode.

Step 3: Edit the addons_def.json.sample file

Edit the file "<INSTALL_DIR>\addons\addons_def.json.sample" as follows:

1. Locate the following section:

```
{
  "addons": [{
    "name": "MyPasswordProvider",
    "type": "STARTUP",
    //"lib_path": "C:\\Program Files\\Attunity
Replicate\\addons\\samples\\MyPasswordProvider\\MyPasswordProvider.dll",
    //"lib_path": "/opt/attunity/replicate/addons/samples/MyPasswordProvider.so",
    "init_function": "ar_addon_init"
  }]
}
```

2. Edit the section as required, where:

- `name` is the logical name of the DLL (can be any name). If you do not use the `lib_path` parameter to specify the DLL path (see below), then the DLL file must reside in <INSTALL_DIR>\addons<addon_name>. On Linux, the <addon_name> folder needs to be created manually and should either be the default addon name (*MyPasswordProvider*) or its new name (if you changed it).
- `type` is an optional parameter that specifies when to load the DLL. The only value that is currently supported is `STARTUP`.
- `lib_path` is the full path of your DLL (e.g. *C:\Vault\MyVaultProvider.dll*). This is required only if the DLL does not reside in <INSTALL_DIR>\addons<addon_name> (as mentioned in the description of the name parameter above).
- `init_function` is the function name, as it appears in the C file used to generate the DLL.

3. Save the file as *addons_def.json*.

Step 4: Configure the endpoints

Any fields where you want the secret to be retrieved from your vault should be configured as described below.

Edit the field as follows:

1. In the appropriate field, enter the following:
`Lookup: :your_lookup_key`
where *your_lookup_key* is the Replicate endpoint identifier specified in your vault access code.
2. Save your settings.
3. Click **Test Connection** to verify that a connection to the endpoint can be established.
4. Run the task.

4.15 Controlling execution of user-defined commands

Several endpoints support running user-defined commands for pre/post-processing files. In the [Operating System Level Credentials](#) tab shown in SERVER view, you can provide user credentials for user-defined commands that needs to be executed at operating system level. By default, such commands are executed

under the Replicate Server service account. This may constitute a security risk, as it allows any Replicate user with Admin or Designer permissions to specify user-defined commands that could lead to a full compromise of the server.

Note that when Replicate Server is installed on Linux, the *External utilities* (page 1277) tab will be hidden as the potential security risk is relevant to Windows only.

You can also determine whether to allow user-defined commands to be executed at all.

To do this:

1. On the Replicate Server machine, open the `<PRODUCT_DIR>\bin\repctl.cfg` file and set the **enable_execute_user_defined_commands** parameter to "true" or "false" (the default) as required.
2. If **Another account** is selected in the [Operating System Level Credentials](#) tab and you [changed the default Attunity Replicate Server service Log On user](#):
 - a. For both admin and non-admin users:
Add the user specified in the Attunity Replicate Server service **Log On** tab to the "Replace a process level token" policy in the "Local Security Policy" settings.
 - b. For non-admin users only:
 - i. Grant the user full control over the Replicate **data** folder.
The default location of the **data** folder is `C:\Program Files\Qlik\Replicate\data`
 - ii. Grant the user full control over the folder containing the user-defined command to be executed.



*The user must be the same as the non-admin user specified in the Attunity Replicate Server service **Log On** tab.*

3. Restart the Qlik Replicate services.

Executing operating system commands as a different user

The load utility of the following target endpoints can run via user-provided credentials instead of the LocalSystem user:

- Microsoft APS PDW
- Google Cloud BigQuery
- IBM Netezza
- PostgreSQL-based

5 Working with Qlik Replicate endpoints

Qlik Replicate can replicate data from the types of endpoints listed in *Supported Platforms and Endpoints* (page 1313). A Replicate endpoint can be either a source or a target. A source endpoint contains the original data (the data you want to copy). A target endpoint is where the replicated data is stored. Source and target can be completely different endpoints.. There is no need to install any additional software other than Replicate. You can also use CDC Agents in Attunity Replicate Connect (ARC) as source data for replication tasks.

For a list of supported endpoint versions, see *Supported Platforms and Endpoints* (page 1313).

In this section:

- *Using ARC CDC agents as endpoints* (page 81)
- *Replicate data types* (page 81)
- *Supported DDL statements* (page 83)
- *Configuring Replicate to automatically replace the user-entered password* (page 85)

5.1 Using ARC CDC agents as endpoints

In a replication project, you can use both relational and non-relational endpoints supported by Attunity Replicate Connect (ARC).



ARC CDC Agents can be used for capturing changes (CDC) only.

Relational and Non-relational endpoints

Relational Endpoints	Non-Relational Endpoints
SQL/MP	HP NonStop Enscribe
-	RMS
-	VSAM
-	IBM IMS

For information on how to work with ARC, see *Using ARC CDC solutions in Qlik Replicate* (page 185).

5.2 Replicate data types

Qlik Replicate converts source data to its own data type. For data that Replicate cannot convert, it returns an error.

To see how a data type is mapped from source to target:

- See the chapter for the source endpoint you are using. In the section on data types, see the mapping table to see the Qlik Replicate data type.

5 Working with Qlik Replicate endpoints

- See the chapter for the target endpoint you are using. In the section on data types, see the mapping table to see how the Replicate data type maps to the target.

For example, when replicating data from an Oracle source endpoint to a Microsoft SQL Server target endpoint, Replicate first converts the Oracle data type BINARY to the Replicate data type BYTES. BYTES maps to the Microsoft SQL Server data type VARBINARY (Length).

The following table describes the Qlik Replicate data types. Some data types have precision and scale information that applies to them.

Replicate data type descriptions

ReplicateData Types	Description
STRING	A character string
WSTRING	A double-byte character string
BOOLEAN	A Boolean value
BYTES	A binary data value
DATE	A date value: Year, Month, Day
TIME	A time value: Hour, Minutes, Seconds Only the following format is supported: HH:MM:SS
DATETIME	A timestamp value: Year, Month, Day, Hour, Minute, Second, Fractional Seconds The fractional seconds have a maximum scale of 9 digits. Year can have a maximum value of 32767. Only the following format is supported: YYYY:MM:DD HH:MM:SS.F(9)
INT1	A one-byte, signed integer
INT2	A two-byte, signed integer
INT4	A four-byte, signed integer
INT8	An eight-byte, signed integer
NUMERIC	An exact numeric value with a fixed precision and scale
REAL4	A single-precision floating-point value
REAL8	A double-precision floating-point value
UINT1	A one-byte, unsigned integer
UINT2	A two-byte, unsigned integer
UINT4	A four-byte, unsigned integer

ReplicateData Types	Description
UINT8	An eight-byte, unsigned integer
BLOB	Binary Large Object
CLOB	Character Large Object
NCLOB	Native Character Large Object

For more information, see **LOB support** in **Task Settings/Metadata** (page 1166).

5.3 Supported DDL statements

Qlik Replicate automatically changes the metadata of the target table to reflect DDL statements performed on the source endpoint.

Supported DDL statements include:

- Create table
- Drop table
- Rename table
- Add column
- Drop column
- Rename column
- Change column data type

For information about supported DDL statements for a specific endpoint, see the chapter describing that endpoint. For more information about DDL settings, see *Apply Changes Settings* (page 1174). See also *Limitations and considerations* (page 84) below.

How Replicate handles DDL changes

When DDL changes occur, Replicate:

1. Captures ALTER TABLE DDLs from the transaction log without identifying the DDL type (ADD/DROP/MODIFY COLUMN).
2. Reads the new table metadata from the source backend.
3. Compares the previous table metadata with the new table metadata in order to determine the change. Note that a single change may include multiple DDL operations performed on the backend.
4. Uses the new table metadata to parse the subsequent DML events.

Limitations and considerations

When capturing DDL changes, the following limitations apply:

- When a rapid sequence of operations occurs in the source database (for instance, DDL>DML>DDL), Replicate might parse the log in the wrong order, resulting in missing data or unpredictable behavior. To minimize the chances of this happening, best practice is to wait for the changes to be applied to the target before performing the next operation.

As an example of this, during change capture, if a source table is renamed multiple times in quick succession (and the second operation renames it back to its original name), an error that the table already exists in the target database might be encountered.

- If you change the name of a table used in a task and then stop the task, Replicate will not capture any changes made to that table after the task is resumed.
- Renaming a source table while a task is stopped is not supported.
- Reallocation of a table's Primary Key columns is not supported (and will therefore not be written to the [DDL History Control Table](#)).
- When a column's data type is changed and the (same) column is then renamed *while* the task is stopped, the DDL change will appear in the [DDL History Control Table](#) as “Drop Column” and then “Add Column” when the task is resumed. Note that the same behavior can also occur as a result of prolonged latency.
- CREATE TABLE operations performed on the source while a task is stopped will be applied to the target when the task is resumed, but will not be recorded as a DDL in the [attrep_ddl_history Control Table](#).
- Operations associated with metadata changes (e.g. ALTER TABLE, reorg, rebuilding a clustered index, etc.) may cause unpredictable behavior if they were performed either:
 - During Full Load
 - OR-
 - Between the [Start processing changes from](#) timestamp and the current time (i.e. the moment the user clicks **OK** in the **Advanced Run Options** window).

Example:

IF:

The specified [Start processing changes from](#) time is 10:00 am.

AND:

A column named **Age** was added to the **Employees** table at 10:10 am.

AND:

The user clicks **OK** in the **Advanced Run Options** window at 10:15 am.

THEN:

Changes that occurred between 10:00 and 10:10 may result in CDC errors.



In any of the above cases, the affected table(s) must be reloaded in order for the data to be properly replicated to the target.

- The DDL statement `ALTER TABLE ADD/MODIFY <column> <data_type> DEFAULT <>` does not replicate the default value to the target and the new/modified column is set to NULL. Note that

this may happen even if the DDL that added/modified the column was executed in the past. If the new/modified column is nullable, the source endpoint updates all the table rows before logging the DDL itself. As a result, Qlik Replicate captures the changes but does not update the target. As the new/modified column is set to NULL, if the target table has no Primary Key/Unique Index, subsequent updates will generate a "zero rows affected" message.

- Modifications to TIMESTAMP and DATE precision columns will not be captured.

5.4 Configuring Replicate to automatically replace the user-entered password

To prevent illicit database activity by unauthorized third-parties, Replicate can be configured to automatically replace the user-entered password with a strong random password.

This feature is currently supported with the following endpoint types:

- Microsoft SQL Server
- Microsoft Azure SQL Database
- Oracle



This feature cannot be used when the user name is "sa".



Clicking the "Test Connection" button will verify the connection using the original password. The password will be automatically changed the first time the task runs.

To utilize this feature, the password must be defined both in the Replicate endpoint settings *and* on the actual database, in the following format:

```
replace:your_password
```

Example:

```
replace:k$RJdg7!S&ib
```

Defining multiple endpoints to use the same automatically changed password

In Qlik Replicate, more than one endpoint may be configured to connect to the same database server.

To allow multiple endpoints to use the same (automatically changed) credentials, the password in one of the endpoints needs to be defined. Then, each of the other endpoint connections needs to be configured to reference that endpoint.



The following rules apply:

- A source endpoint cannot reference a target endpoint, only another source endpoint.
- A target endpoint cannot reference a source endpoint, only another target endpoint.
- An endpoint cannot reference another endpoint that uses a different database server.

To configure an endpoint to use the automatically changed credentials of another endpoint:

1. In the **User name** field, enter the user name in the following format:

```
ref:endpoint_name
```

Where `endpoint_name` is the name of the endpoint connection whose password was automatically changed.

2. In the **Password** field, specify the password *before* it was automatically changed and without the "replace" prefix.

Example:

If the original password is:

```
replace:54lakrfgnier3!
```

Specify:

```
54lakrfgnier3!
```

6 Using the Qlik Replicate Console

The Qlik Replicate Console is a Web-based application that runs in most browsers (for information on supported browsers, see *Supported browsers (page 1327)*). You can connect from any computer to the Replicate Server.

This section describes the user interface elements in the Replicate Console.

In this section:

- *Opening the Qlik Replicate console (page 87)*
- *Tasks view (page 91)*
- *Server view (page 96)*
- *List actions (page 96)*

6.1 Opening the Qlik Replicate console

You browse to the Qlik Replicate Console using a supported Web browser from a computer in the same network as the computer on which the Qlik Replicate Server is installed. For information on supported browsers, see *Supported browsers (page 1327)*.

You can open the Console from the **Start** menu of the computer where you installed Qlik Replicate.

To enable and control access to Qlik Replicate, you can create user roles as described in *User permissions (page 1274)*.

To open Qlik Replicate

- Click **Start** and from the **All Programs** section point to **Qlik Replicate** and select **Qlik Replicate Console**.



When you connect to the Qlik Replicate Console, your browser will prompt you for a username and password. The username and password that you need to specify depends whether Replicate Server is installed on Windows or Linux.

- **Qlik Replicate Server on Windows:** Your domain username and password.
- **Qlik Replicate Server on Linux:** Either specify your PAM credentials or, if PAM is not set up in your environment, specify `admin` as your username and the Replicate Server password as your password.

*For information on setting the Replicate Server password, see *Security considerations (page 60)*.*

*For information on PAM prerequisites, see *Configuration 3: Replicate UI Console and Replicate server running on Linux (page 90)*.*

Accessing Qlik Replicate from a remote computer

You can access Qlik Replicate from any computer in your network.

Accessing the Qlik Replicate Console

To access the Qlik Replicate Console from a remote computer, type the following address in the address bar of your Web browser:

Qlik Replicate Server on Windows:

`https://computer-name/attunityreplicate`

Qlik Replicate Server on Linux:

`https://computer-name:<port>/attunityreplicate`

Where `<computer name>` is the name or IP address of the computer where the Qlik Replicate Server is installed and `<port>` is the C UI Server port (3552 by default). For more information on the C UI Server component, see *Qlik Replicate UI server configurations (page 89)*.



*The person logged in to the computer where you are accessing the Console must be an authorized Qlik Replicate user. For more information, see *User permissions (page 1274)*.*

Viewing the current connection values URL

To see the current connection values:

1. Open a command prompt as an administrator and change the working directory to the Replicate **bin** directory.
2. Run the following command:

```
repuictl configuration show
```

The output will be similar to this:

```
--address ""
--http_port "80"
--https_port "443"
--root_url "attunityreplicate"
--user_timeout "5"
--domain "mydomain.com"
--ad_default_container ""
--authentication_method "SSO"
--static_http_headers
--rest_http_headers
--use_strict_same_site_cookie "False"
```



An empty address (`--address ""`) means that the current computer name will be used.

See also:

- [Setting up Qlik Replicate UI Server to work with a remote Qlik Replicate Server \(page 90\)](#)

Qlik Replicate UI server configurations

You can either install Qlik Replicate on a single machine or on two separate machines. The possible configurations for installing Qlik Replicate on two separate machines are described below.

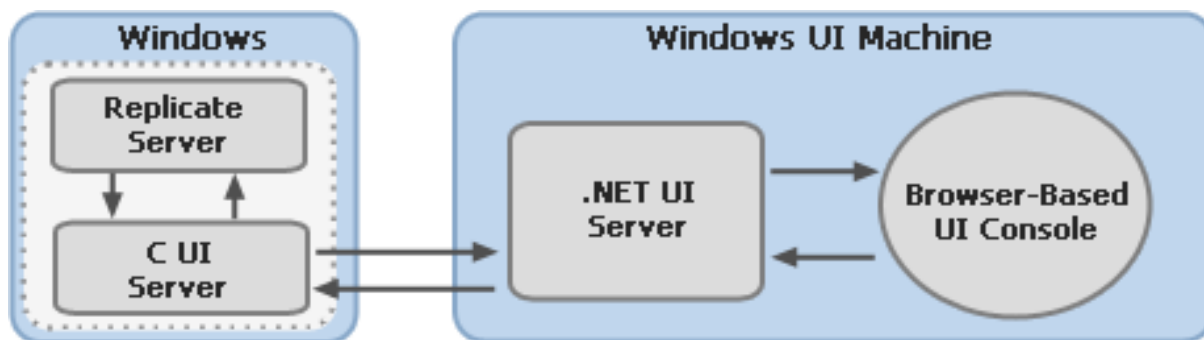


When the Qlik Replicate .NET UI Server is running on one machine and the Qlik Replicate Server is running on another, the Qlik Replicate Server password must be the same on both machines. The password is used during the SSL handshake to establish a secure connection between the participating machines.

For information on setting the password, see [Changing the server password \(page 69\)](#).

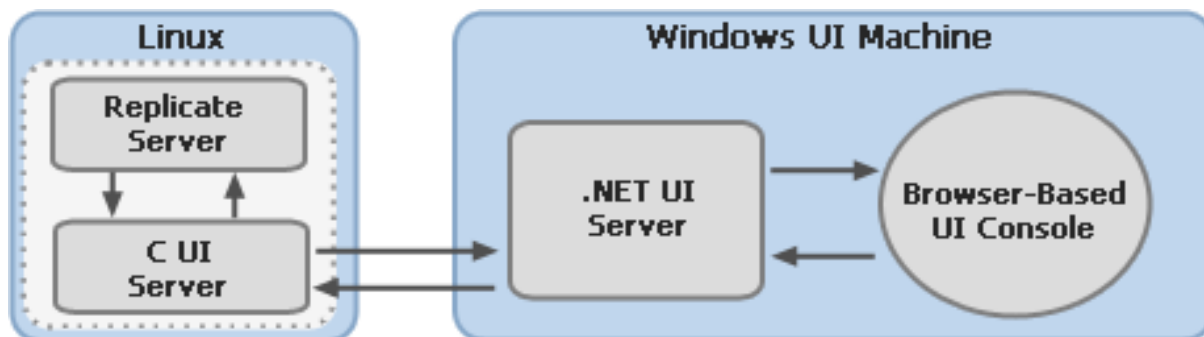
Configuration 1: Replicate Server running on Windows

In this configuration, the Replicate Console component and the Replicate Server components are running on two separate Windows machines.



Configuration 2: Replicate server running on Linux

In this configuration, the Replicate Console component and the Replicate Server components are running on two separate machines - the former on Windows and the latter on Linux.



Configuration 3: Replicate UI Console and Replicate server running on Linux

In this configuration, the UI Console and the Web server (Qlik Replicate Server) are hosted on two separate Linux machines, though it is also possible to install them on a single machine.

Note that in such a configuration, the ability to assign different roles (as described in *User permissions (page 1274)*) is not supported. In other words, all users will have the admin role.

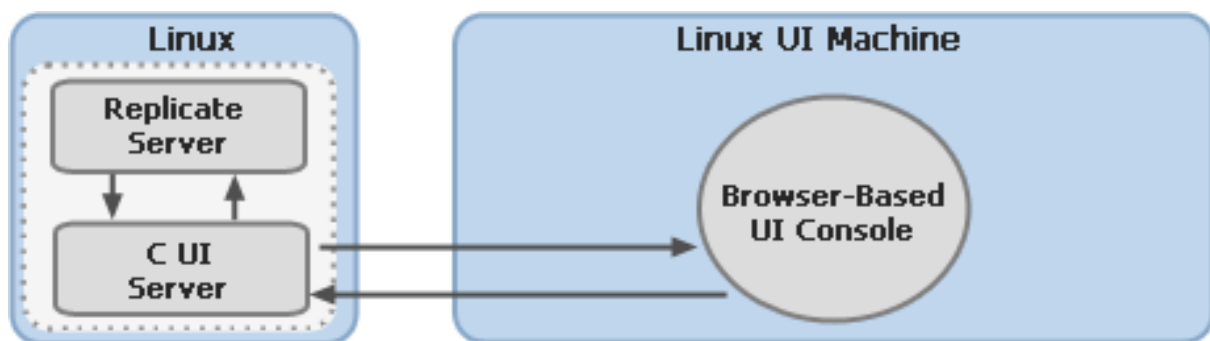
PAM prerequisites

To establish a secure connection using PAM, make sure that the following prerequisites have been met:

- The Attunity user or group (or the user/group set during the installation) must be granted permission to read the file: `etc/shadow`. Note that this prerequisite is only required when Qlik Replicate is installed on two machines.
- Edit the `repctl.cfg` file and modify the path to the fully qualified name of the `libpam.so.0` library if required.

Example:

```
"login_pam_libpam_full_path": "/lib64/libpam.so.0",  
"login_pam_service_name": "system-auth"  
}
```



Setting up Qlik Replicate UI Server to work with a remote Qlik Replicate Server

If the Qlik Replicate Server is installed remotely (that is, not on the Qlik Replicate UI Server machine), do the following:

1. On the console machine, open a command prompt as an administrator and change the working directory to the Replicate **bin** directory.
2. Run the following command:

```
repuctl configuration set_replicate --replicate_host host [--replicate_port port]
```

where:

- `host` is the IP address or host name of the Replicate Server machine.
 - `port` is the port of the Replicate Server machine. If you omit this parameter, the default port (3554) will be used.
3. Open the Windows Services console and restart the **Qlik Replicate UI Server** service.
 4. Connect as described in *Opening the Qlik Replicate console (page 87)*.

Viewing the configuration

You can view the current connection configuration.

To do this:

1. Open a command prompt as an administrator and change the working directory to the Replicate **bin** directory.
2. Run the following command:

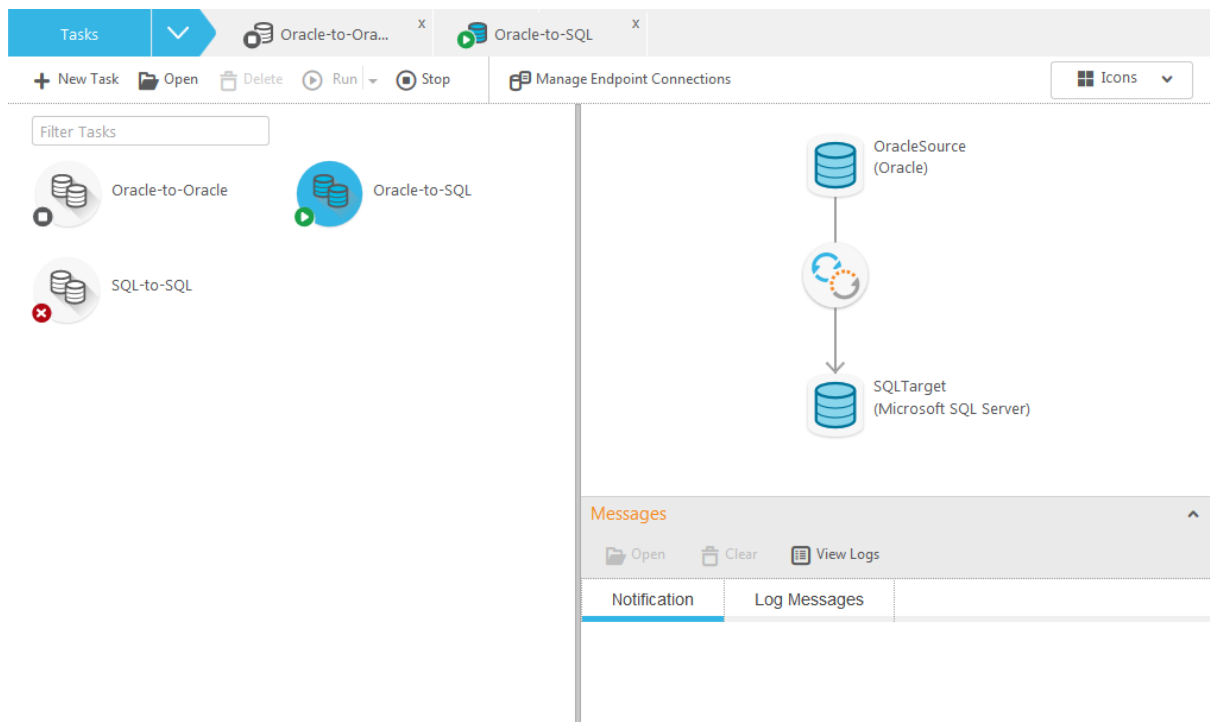
```
repuictl configuration show_replicate
```

See also:

- *Managing user permissions (page 1275)*

6.2 Tasks view

The **Tasks** view is the default view that opens when you open Qlik Replicate for the first time, as shown in the following image:




It lists all replication tasks you have defined. You use this view to view, edit, run, and delete tasks, or to create new tasks.

This view includes the following elements:

- **Toolbar** running along the top. It includes buttons that let you create a new task, open, delete, run, or stop an existing task, configure advanced run options, and manage endpoint connections. See also *Adding tasks (page 110)*.

- **Tasks** already defined in the system, listed in the left pane.

You can view tasks in:

- **Icons** view, where each icon indicates the current state of the tasks. See the *Tasks view (page 91)* table for more information.
- **Details** view, which displays a table with additional information about each task including their current state. Note that the state icons are the same as described in the *Tasks view (page 91)* table, but without the  part of the icon.

To toggle between these views, you can select **Icons** or **Details** from the drop-down list in the top right of the Console.

For information about creating a task, see *Defining and managing tasks (page 109)*.

The Console displays each open task on its own tab along the top. For more information, see *Viewing specific tasks (page 93)*.



- **Endpoints map** in the right pane, which illustrates the endpoints for the task selected in the left pane. Any notifications (if defined) and log messages will be shown in the **Messages** pane below the map.
- **Messages pane** below the endpoints diagram on the right. This pane includes a **Notifications** tab for progress messages and a **Log Messages** tab for warnings and error messages issued when Replicate encounters a problem. For more information, see *Accessing task messages and notifications (page 1212)* and *Creating a new notification (page 1240)*.




To access the Tasks view:

- Select **Tasks** from the drop-down list in the top left, below the Qlik Replicate logo.

The following table shows examples of task icons.

Task icons

Icon	Description
	Indicates that the task has not been run yet.
	Can be one of the following: <ul style="list-style-type: none"> • Manually stopped by the user • Stopped due to the task definition (Full Load settings) • Stopped by the Scheduler

Icon	Description
	Indicates that the task has stopped due to an error. When you select the task, Replicate displays a list of errors on the Log Messages tab at the bottom right of the console.
	Indicates that the task has stopped due to a recoverable error. When you select the task, Replicate displays a list of errors on the Log Messages tab at the bottom right of the console.
	Indicates that the task is running.

Viewing specific tasks

From the **Tasks** view, you can drill down to an individual task, provided you have already created at least one task (see *Defining and managing tasks (page 109)* for more information). Two modes display different sets of information for each task:

- *Designer mode (page 93)*: Default mode when you open a task. Here you define endpoints, select tables, modify table settings (including filters and transformations), and create global transformation rules.
- *Monitor mode (page 95)*: Here you view replication task activities in real time, along with log messages and notifications.

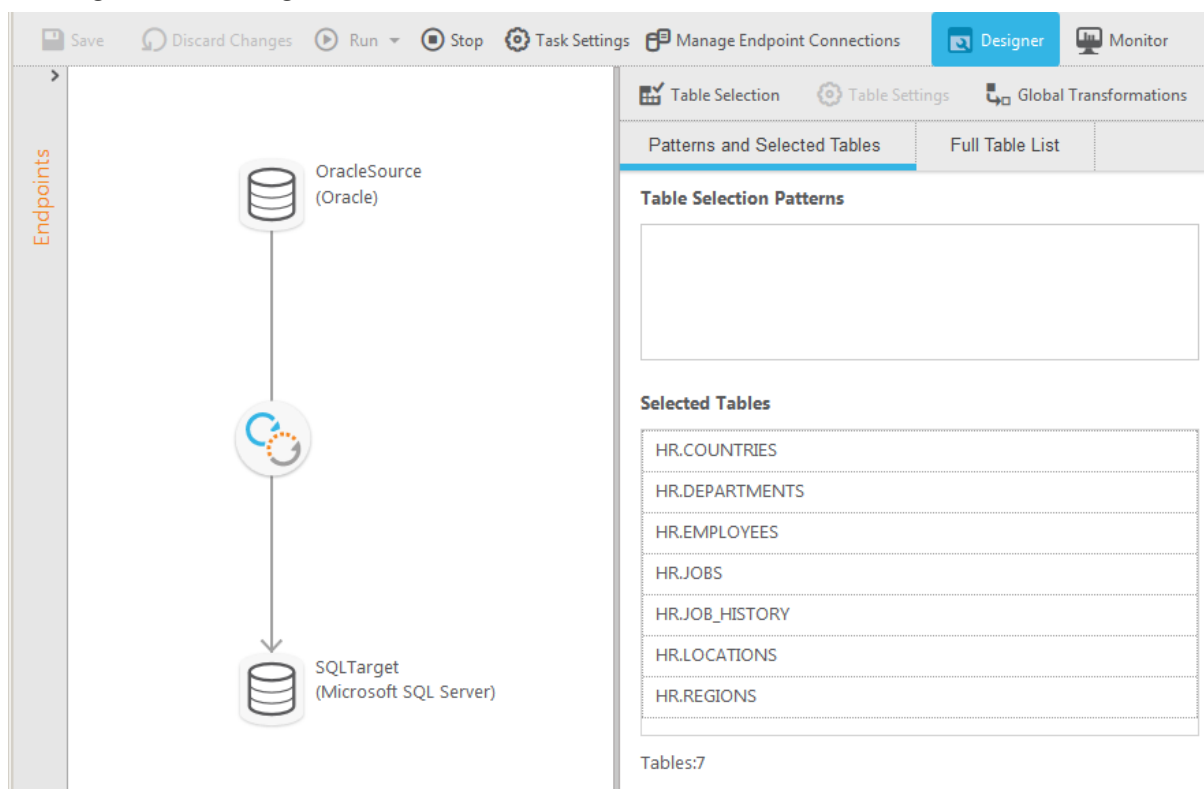
To view a specific task:

1. In the **Tasks** view, select the task you want to work with.
The right pane displays the task diagram on the right side of the page.
2. On the **Tasks** view toolbar, click **Open**.

Designer mode

In **Designer** mode, you define endpoints, select tables to be replicated, modify table settings (including filters and transformations), and create global transformation rules. This is the default mode when you open a task.

Viewing a task in Designer mode



The **Designer** mode includes the following elements:

- **Endpoints list:** Lists the source and target endpoint connections that you added to Qlik Replicate. For more information, see *Defining and managing endpoints (page 114)*. The figure shows the Endpoints List in a collapsed state, hiding the endpoints. To expand the list, click the right arrow at the top or anywhere below it. To close the panel, click the left arrow.
- **Endpoints map:** Illustrates the connection between the source and target endpoints for the task. The round icon between the endpoints represents the task type, which can indicate Full Load only, Full Load and Apply Changes, or Apply Changes only.
- When you create a task, you can drag the endpoints to the source and target drop spots as required. For more information, see *Adding a source and target endpoint to a task (page 117)*.
- **Monitor and Designer buttons:** Lets you switch between **Monitor** mode and **Designer** mode. See also *Monitor mode (page 95)* and *Monitoring and controlling replication tasks (page 1216)*.
- **Run button:** Lets you run the task at hand.
- **Task Settings button:** Opens the **Task Settings** dialog box. For more information, see *Task Settings (page 1165)*.
- **Manage Endpoint Connections button:** Lets you view the endpoints defined, edit them, or add new endpoints. For more information, see *Defining and managing endpoints (page 114)*.
- **Select and Define Tables:** Lets you select the tables you want to include in your replication task. In addition, you can use transformation and filter operations to create new tables or to replicate parts of tables. For more information, see *Selecting tables and/or views for replication (page 118)*, *Using filters (page 1092)*, and *Defining transformations for a single table/view (page 1084)*.

- **Global Transformations option:** Lets you create transformations for all tables in a task. For more information, see *Starting the Global Transformation Rules wizard (page 1107)*.

To display a task in Designer mode:

- On the right side of the toolbar, click **Designer**.

Monitor mode

In **Monitor** mode, you view the replication task activities in real time.

Viewing a task in Monitor mode

Table Name	Status	Estimated Count	Count
HR.COUNTRIES	Completed	25	25
HR.DEPARTMENTS	Completed	27	27
HR.EMPLOYEES	Completed	107	107

The **Monitor** mode includes the following elements:

- **Run button:** Lets you run the task at hand.
- **Manage Endpoint Connections button:** Lets you view the endpoints defined, edit them, or add new endpoints. For more information, see *Defining and managing endpoints (page 114)*.
- **Monitor and Designer buttons:** Switch between **Monitor** mode and **Designer** mode. See also *Monitoring and controlling replication tasks (page 1216)*, *Designer mode (page 93)*, *Defining and managing tasks (page 109)*.
- **Tools list:** Provides access to history, log management, and status information.
- **Change Processing/Full Load tabs:** Lets you select the information you want to focus on. By default, Replicate displays the **Full Load** view (also shown in the figure).

- **Task Map:** Illustrates the connection between the source and target endpoints for the task. The round icon between the endpoints represents the task type, which can indicate Full Load only, Full Load and Apply Changes, or Apply Changes only.
- **Messages pane:** Displays notifications and logging messages. For more information, see *Accessing task messages and notifications (page 1212)*.

To display a task in Monitor mode:

- On the right side of the toolbar, click **Monitor**.

6.3 Server view

SERVER view lets you view and configure the Qlik Replicate Server settings.

To switch to **SERVER** view:

- From the drop-down list in the top left corner of the console (below the product logo) select **Server**.

For information on configuring server settings, see *Qlik Replicate server settings (page 1239)*.

6.4 List actions

The following table describes the various list actions you can perform. Note that, depending on the list type, some of the actions may not be available.

List actions

To	Do This
Sort ascending or sort descending	Right click the desired column and select one of the sorting options as required.
Restore the default sorting order	Right click any of the column headings and select Default Sorting .
Export the list to a TSV file	The following lists can be exported: tasks, messages, selected tables, and processed tables (in Monitor view). Either click the Export to TSV button above the list or right-click any of the column headings and select Export to TSV . Choose where to save the file and then click Save .
Add or remove columns	Right click any of the column headings and select Column Settings . Then add or remove columns as required.
Hide a column	Right click the desired column and select Hide Column .

7 Getting started: A Qlik Replicate tutorial

This section guides you through setting up a basic replication task to move data from an Oracle source to a Microsoft SQL Server target.

In this section:

- *What you need (page 97)*
- *Open the Qlik Replicate console (page 97)*
- *Add an Oracle endpoint as a source (page 98)*
- *Open the Qlik Replicate console (page 97)*
- *Add a Microsoft SQL Server database as a target (page 99)*
- *Define a replication task (page 102)*
- *Run and monitor the replication task (page 107)*
- *View the replicated tables in Microsoft SQL Server (page 108)*

7.1 What you need

For this tutorial, you need the following:

- Qlik Replicate installed on a computer in your network
- For the Oracle source:
 - Access to the HR schema tables that are part of the Oracle database installation



If these tables are not available, contact your Oracle database administrator.

- `system/<password>` for an admin user
- For the target: A Microsoft SQL Server database with the default tempdb system database (used to store the target tables).
This can be installed on your local computer.
- For the Qlik Replicate Console, one of the *Supported browsers (page 1327)*.

See also: *Installing Qlik Replicate (page 39)* *Windows permissions (page 41)*

7.2 Open the Qlik Replicate console

From the Windows Start menu, select **All Programs > Qlik Replicate > Qlik Replicate Console**.



You can access Qlik Replicate from any computer in your system.

To access the Console from a remote computer, type the following address in the address bar of your Web browser:

`https://<computer name>/attunityreplicate`

where <computer name> is the name or IP address of the computer (including the Windows domain name) on which the Qlik Replicate Server is installed.



The person logged in to the computer hosting the Console must be an authorized Qlik Replicate user. For more information, see *User permissions* (page 1274).

7.3 Add an Oracle endpoint as a source

This task guides you through adding and configuring an Oracle endpoint as the source database. This is the database from where you want to replicate data.

To add an Oracle source database:

1. In Task view, click **Manage Endpoint Connections**.
The **Manage Endpoint Connections** dialog box opens.
2. Click **New Endpoint Connection**.
3. Provide the following information:
 - **Name:** Type **OracleSource**.
 - **Description:** Optionally, enter a description or leave blank.
 - **Role:** Select **Source**.
 - **Type:** Select **Oracle**.
 - **Connection string:** Enter the connect string to the Oracle database you work with, in any Oracle format.
For example, if you connect to an Oracle database on a computer called **tomato** using the default Oracle port and service name, the connect string looks like this:
`tomato:1521/orcl`
 - **User Name:** Enter the user name for the Oracle database you work with.
The default user name is **SYSTEM**.
 - **Password:** Enter the password for the Oracle database you work with.
The default password is **manager**.

The screenshot shows the 'General' tab of a configuration window. The 'Name' field is 'OracleSource'. The 'Description' field is empty. The 'Role' is 'Source' and the 'Type' is 'Oracle'. The 'Connection string' is 'tomato:1521/orcl', 'User name' is 'SYSTEM', and 'Password' is masked with four dots. At the bottom, there are buttons for 'Test Connection', 'Save', 'Discard Changes', and 'Close'.

Name:	OracleSource
Description:	
Role:	Source
Type:	Oracle
Connection string:	tomato:1521/orcl
User name:	SYSTEM
Password:	****

Buttons: Test Connection, Save, Discard Changes, Close

4. Click **Test Connection** to verify the information you entered and the availability of the database.
5. Click **Save** to add the database.

You can also set advanced settings for the Oracle database, but this is beyond the scope of this tutorial. For more information, see *Setting advanced connection properties using Oracle LogMiner (page 442)*.

For information on adding other types of databases, see the chapter for the required database. For a list of supported databases, see *Supported Platforms and Endpoints (page 1313)*.

7.4 Add a Microsoft SQL Server database as a target

This task guides you through adding and configuring a Microsoft SQL Server endpoint as the target database connection. This is the database to where you want to replicate data.

To add a Microsoft SQL Server target endpoint:

1. In Tasks view, click **Manage Endpoint Connections**.
The Manage Endpoint Connections dialog box opens.
2. Click **New Endpoint Connection**.
3. Provide the following information:
 - **Name:** Type `sqlserver_target`.
 - **Description:** Optionally, enter a description or leave blank.
 - **Role:** Select **Target**.
 - **Server name:** Enter the name of the computer where your Microsoft SQL Server database is installed.
For example, if you connect to a Microsoft SQL Server database on a computer called `bee`, enter `bee`.
 - Select one of the following:
 - **Windows authentication** if your Microsoft SQL Server database is configured to accept Windows authentication.
 - **Microsoft SQL Server authentication** if your Microsoft SQL Server database is not configured to accept Windows authentication. In this case, you also need to provide a valid user name and password.
 - **Database name:** Enter **tempdb**, which is the name of the database to where you are going to replicate data. If you created a new database for this purpose, enter the name of that database.

The screenshot shows a configuration dialog with two tabs: 'General' (selected) and 'Advanced'. The 'General' tab contains the following fields and options:

- Name:** sqlserver_target
- Description:** (empty text area)
- Role:** Source Target
- Type:** Microsoft SQL Server (dropdown menu)
- Server name:** bee01-win7
- Authentication:** Windows authentication SQL Server authentication
- User name:** (disabled text field)
- Password:** (disabled text field)
- Database name:** tempdb (text field) with a **Browse...** button to its right.

At the bottom of the dialog, there are four buttons: **Test Connection**, **Save**, **Discard Changes**, and **Close** (located in a separate box at the bottom right).

4. Click **Test Connection** to verify the information you entered and the availability of the database.
5. Click **Save** to add the database.

You can also set advanced settings for the Microsoft SQL Server database, but this is beyond the scope of this tutorial. For more information, see *Setting advanced connection properties* (page 364).

For information on adding other types of databases, see the chapter for the required database. For a list of supported databases, see *Supported Platforms and Endpoints* (page 1313).

7.5 Define a replication task

This task guides you through defining a replication task that copies the data from the HR.EMPLOYEES and HR.JOBS tables. It is not mandatory to add a source and a target database prior to this step; you can also do this as part of setting up the replication task.

By default the Oracle database includes the HR schema. You will make a copy of the same tables in your Microsoft SQL Server `tempdb`. The EMPLOYEES and JOBS tables created in Microsoft SQL Server will be identical to the Oracle tables.

For information on how to use Transformations and Filters when creating a replication task, see *Defining transformations for a single table/view* (page 1084) and *Using filters* (page 1092).

Defining a replication task includes the following sub-tasks:

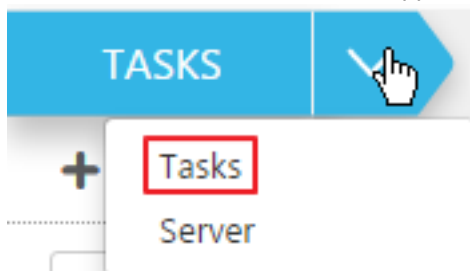
- Add a replication task to the Qlik Replicate console (page 102)
- Add the source and target endpoints to the task (page 104)
- Select tables for the Replication task (page 106)

Add a replication task to the Qlik Replicate console

This task guides you through adding a replication task to the Qlik Replicate Console.

To add a replication task:

1. Make sure **Tasks** is selected in the upper left corner of the Qlik Replicate Console.



2. Click **New Task** to open the New Task dialog box.
3. In the New Task dialog box, in the **Name** field, type `My_Task` and click **OK**.

New Task ✕

Name ⁱ

Description

Replication Profile

Select the profile that best describes your replication use case

Unidirectional


Bidirectional ⁱ

Log Stream ⁱ


Qlik Cloud Landing ⁱ

Task Options


Select the replication options for this task



Full Load



Apply Changes



Store Changes

Perform a full load of target tables, then keep them up-to-date by applying changes.

The Qlik Replicate Console displays the task on a new tab. By default, because the task has not been set up yet, the tab opens in Designer view. The diagram on the left serves as a drop-off point for the

source and target databases you defined previously. The right pane lets you select the tables you want to work with and carry out transformations and filtering operations. For more information, see *Tasks view (page 91)*, *Viewing specific tasks (page 93)*, and *Defining and managing tasks (page 109)*. If needed, you can also change the default task settings. For more information, see *Task Settings (page 1165)*.



See also:

- *Bidirectional replication (page 111)*
- *Using the Log Stream (page 1072)*

Add the source and target endpoints to the task

This section guides you through adding the source and target endpoints to the replication task, which is a simple drag-and-drop operation. In the **Endpoints** tab, the following icons help you distinguish between source and target endpoints:

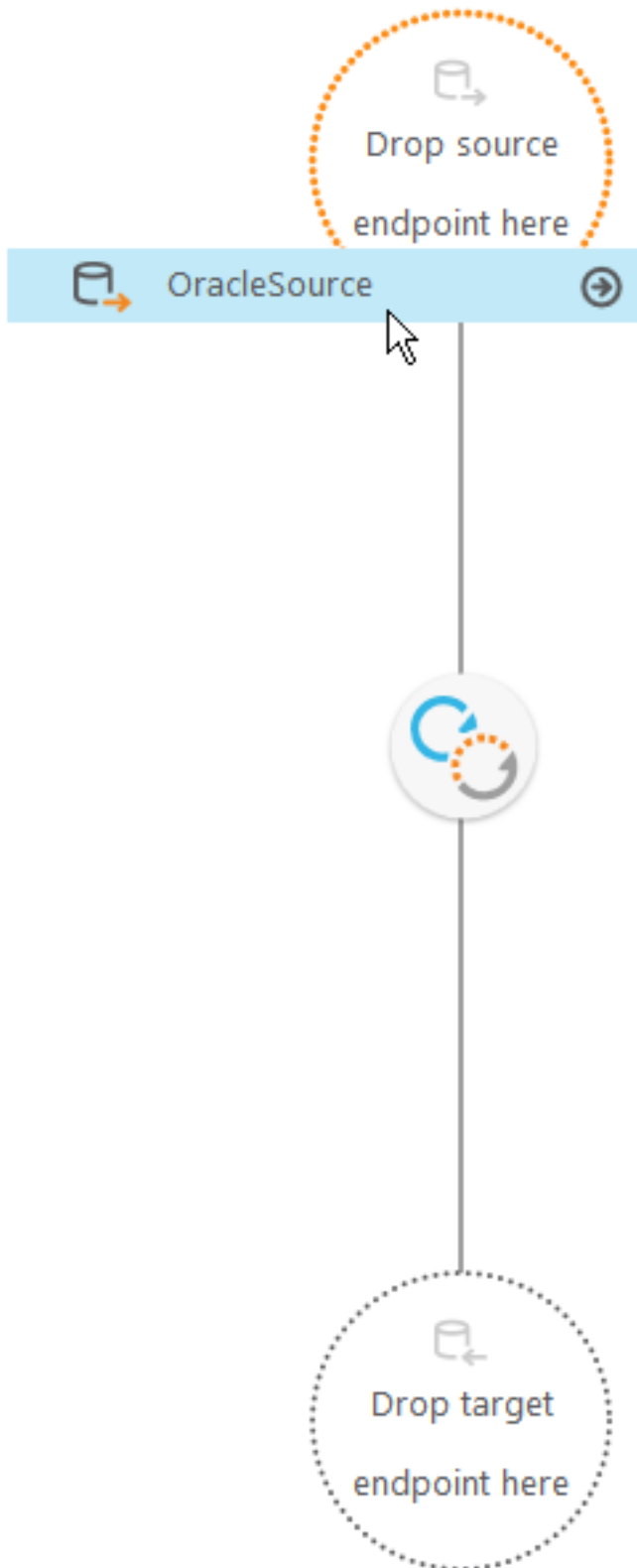
Endpoint icons

Icon	Description
	Source endpoint, which is represented by a database, file, or NoSQL icon, depending on the endpoint type, with an orange arrow pointing away from the source (a database in this example).
	Target endpoint, which is represented by a database, file or NoSQL icon, depending on the endpoint type, with a blue arrow pointing toward the target (a database in this example).

The **Endpoints** pane consists of **All**, **Source**, and **Target** tabs.

To add the source or target endpoints to the task:

1. In the **Endpoints** pane, click the **All** tab.
2. Drag the OracleSource database to the **Drop source endpoint here** area in the endpoints diagram.



3. Drag the `sqlserver_targetdatabase` to the **Drop target endpoint here** area.

Next, you can select the tables from the source database to use in the replication task. For more information, see *Defining and managing tasks* (page 109).

Select tables for the Replication task

After adding the source and target databases, you now need to select which Oracle source tables you want to replicate to the Microsoft SQL Server target.

This task guides you through selecting specific tables (HR.EMPLOYEES and HR.JOBS) from the Oracle source. Replicate takes all of the data from these tables "as is" and copies it to the Microsoft SQL Server target.

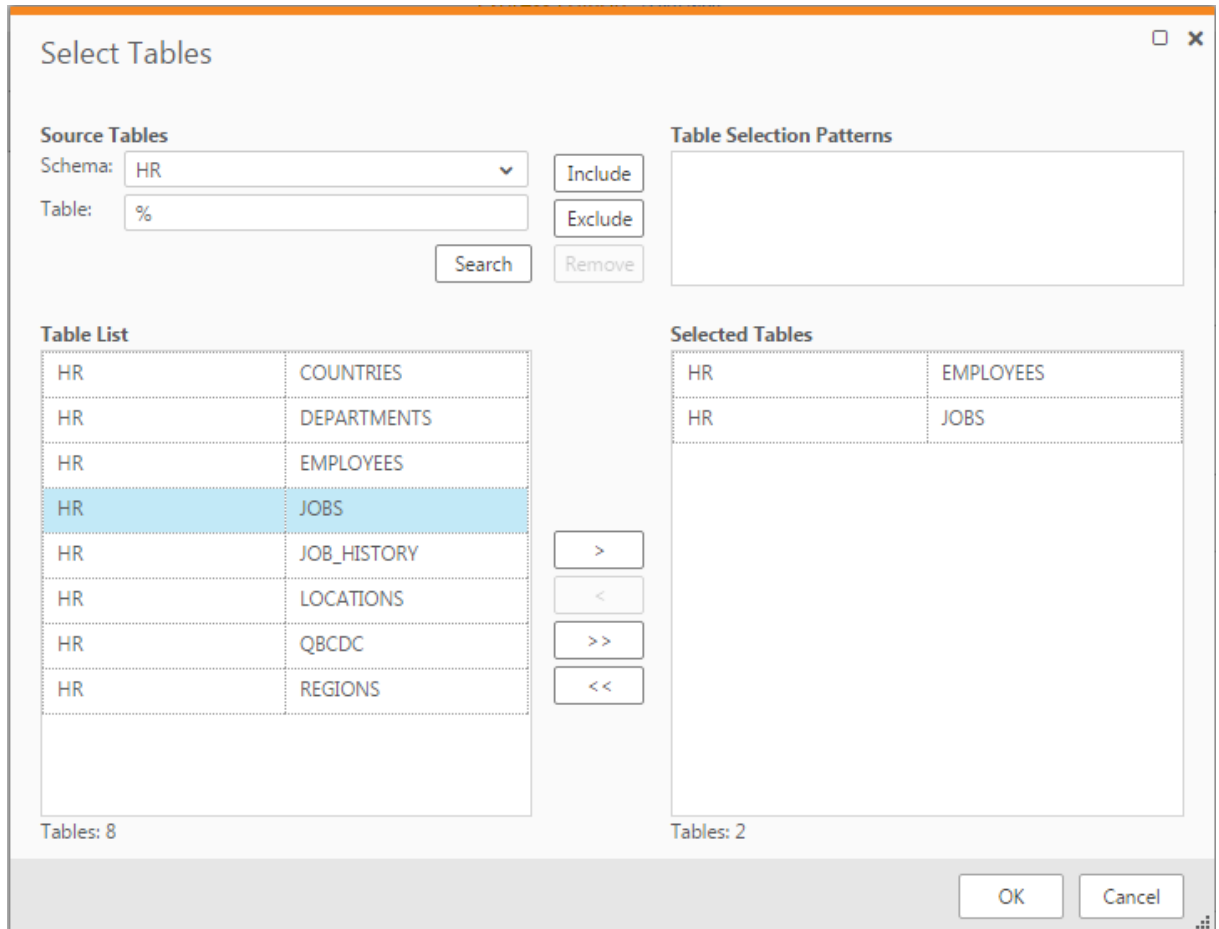


If you need to copy only some of the data to the target database, you need to use a filter. For information, see [Using filters \(page 1092\)](#).

If you need to copy the data into the target using different rows or columns than those in the source, you need to use transforms. For more information, see [Defining transformations for a single table/view \(page 1084\)](#).

To add tables to the replication task:

1. In the right pane of the Qlik Replicate Console, click **Table Selection**. The Select Tables dialog box opens.
2. In the Select Tables dialog box, do the following:
 - From the **Schema** list, select **HR**, and then click **Search**.
 - From the **Table List**, select **EMPLOYEES**, and then click the right arrow to select that table.
 - Repeat these steps for the **JOBS** table.
 - Click **OK**.



3. On the task tab, click **Save**. The task is now ready to run.

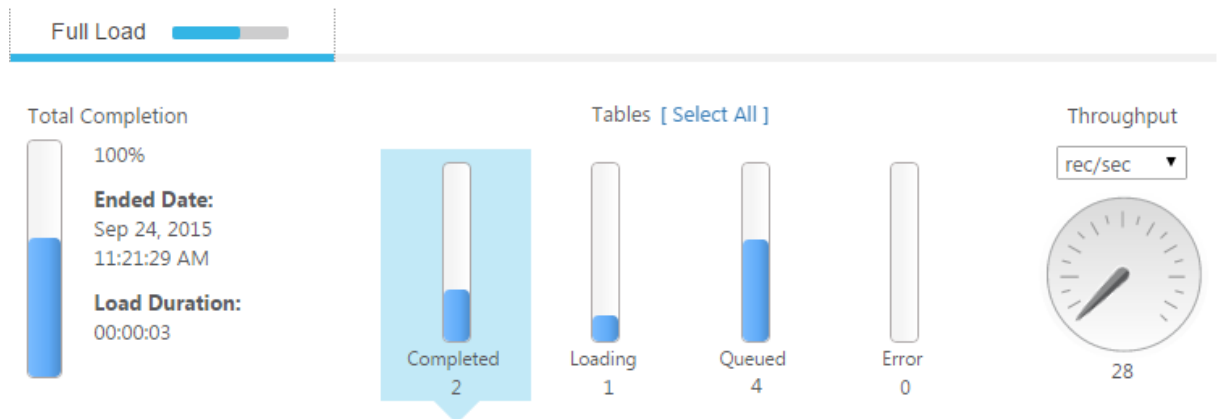
7.6 Run and monitor the replication task

You can now run the replication task and see the results of the replication in real time. This task guides you through running the replication task as a full load and viewing the progress in the Monitor. Additional run options are also available. For more information, see *Using the Run button options (page 1204)*.

To run and monitor the replication task:

1. On the task tab, click **Run**.
The **Starting task** message displays, and the console switches to Monitor view, which includes gauges and graphs on two tabs:
 - **Full Load** tab: Indicates the status progress during the full load process.
 - **Change Processing** tab: Monitors changes that occur after the full load completes.
For information on reading the data presented in these sections, see *Viewing information in the monitor (page 1216)*.

7 Getting started: A Qlik Replicate tutorial



2. Click the **Select All** link above the Tables graphs. Replicate displays a table below the graphs with information about each of the tables being processed in the task.

Tables - All					
Filter By					
Resume Reload					
Table Name	Status	Estimated Count	Count	Elapsed Time	Progress
HR.EMPLOYEES	Completed	107	107	00:00:02	100%
HR.JOBS	Completed	19	19	00:00:02	100%

3. Click the individual bar graphs, such as the Completed graph and the Loading graph, to view additional information.
For information about the data supplied in these tables, see *Monitoring full-load operations (page 1216)*.

7.7 View the replicated tables in Microsoft SQL Server

This task guides you through viewing the `tempdb` database in Microsoft SQL Server. You will see that this database now includes two new tables: `HR.EMPLOYEES` and `HR.JOBS`.

To view the replicated tables in Microsoft SQL Server:

1. From the Windows **Start** menu, go to **All Programs > Microsoft SQL Server > Microsoft SQL Server Management Studio**.
2. In the Object Explorer, find the Microsoft SQL Server target computer you are working with.
3. Expand the **databases** folder for that computer, then expand the **System databases** folder, then expand the **tempdb** database. The `EMPLOYEES` and `JOBS` tables should now appear in the list.
4. Right-click the `EMPLOYEES` table and select **Select Top 1000 Rows**. Check that there is data in the table.
5. Right-click the `JOBS` table and select **Select Top 1000 Rows**. Check that there is data in the table.

8 Defining and managing tasks

This section describes how to design a replication task. To design a replication task, you must first be sure that you have configured at least one source endpoint and one target endpoint to work with Qlik Replicate.



A number of variables affect the amount of tasks that can be run on a single Replicate Server, including the task configuration (e.g. how many tables are being replicated), the size of the source tables and the hardware configuration of the Replicate Server machine. Bearing this in mind, the number of tasks that can be run on a single Replicate Server should not exceed 100 (and may need to be significantly less depending on the aforementioned variables). Best practice is to perform load testing in a Test environment before moving to Production.

In this section:

- [Tasks overview \(page 109\)](#)
- [Adding tasks \(page 110\)](#)
- [Defining and managing endpoints \(page 114\)](#)
- [Adding a source and target endpoint to a task \(page 117\)](#)
- [Selecting tables and/or views for replication \(page 118\)](#)
- [Editing a replication task \(page 125\)](#)
- [Deleting a replication task \(page 125\)](#)
- [Searching for tasks \(page 126\)](#)
- [Migrating tasks \(page 126\)](#)

8.1 Tasks overview

Each instance of a table synchronization activity comprises a task in Qlik Replicate. You define a task using the browser-based Qlik Replicate Console. When defining a task, you specify the following:

- The source and target endpoints
- The source and target tables to be kept in sync
- The relevant source table columns
- The filtering conditions (if any) for each source table as Boolean predicates (in SQLite syntax) on the values of one or more source columns
- The target table columns (optionally), including their data types and values (as expressions or functions over the values of one or more source or target columns, using SQL syntax). If not specified, Replicate uses the same column names and values as the source tables, with default mapping of the source DBMS data types onto the target DBMS data types. Replicate automatically takes care of the required filtering, transformations, and computations during the Load or CDC execution.

When a task is defined, you can activate it immediately. Replicate automatically creates and loads the target tables with the necessary metadata definitions and activates the CDC. Using the Qlik Replicate Console, you can then monitor, stop, or restart the replication process.

Using multiple tasks

You can define and activate several replication tasks at once. This is best if the tasks:

- Have different source tables
- Share some source tables but have different filtering conditions on the source rows
- Update different target tables

Updating the same target table and row by two different replication tasks would not be good practice and may cause unpredictable results.

The different replication tasks work independently and run concurrently. Each has its own Initial Load, CDC, and Log Reader processes.

8.2 Adding tasks

Before you get started with designing the features that you need for a task, you must first define the task's default behavior.

To add a task:

1. In Tasks view, click **New Task**.
The **New Task** dialog box opens.

New Task ×

Name i

Description

Replication Profile
Select the profile that best describes your replication use case

Unidirectional

Bidirectional i

Log Stream i

Qlik Cloud Landing i

Task Options
Select the replication options for this task


Full Load


Apply Changes

Store Changes

Perform a full load of target tables, then keep them up-to-date by applying changes.

2. Enter a name for the task. The name should be descriptive to indicate the purpose of the task. The name cannot exceed 32 characters, contain non-Latin characters, or contain any of the following characters: | \ / : * ? " < >
3. Optionally, enter a description for the task.
4. Choose one of the following replication profiles:
 - **Unidirectional** - Choose to replicate between endpoints for the purpose of Unidirectional.
 - **Bidirectional** - Choose to synchronize records between two endpoints.
For more information, see the instructions on setting up *Bidirectional replication* (page 111).
 - **Log Stream Staging** - Log Stream Staging enables a dedicated Replicate task to save data changes from the transaction log of a single source database and apply them to multiple targets, without the overhead of reading the logs for each target separately.
For more information, see *Using the Log Stream* (page 1072).
5. Select task options:
 - **Full Load**: Click to enable or disable Full Load options for this task.
When full load is enabled, Qlik Replicate loads the initial source data to the target endpoint. By default, a full load is carried out for this task. If you want to change this setting after you begin working with this task, you make the change in the Task Settings, *Full Load* (page 1171) tab.
 - **Apply Changes**: Click to enable or disable Apply Changes (Change Processing).
When this option is enabled, Qlik Replicate keeps the target tables continually up-to-date with any changes made to the source tables. By default, change processing is carried out for this task. You can view the change processing in the Monitor view.
For more information, see *Monitoring change processing operations* (page 1224). If you want to change this setting after you begin working with this task, you make the change in the Task Settings > *Change Processing* (page 1174) tab.

 When the **Bidirectional** replication profile is selected, the **Apply Changes** option cannot be disabled.
 - **Store Changes**: Click this button to enable or disable **Store Changes**.
If this option is enabled, changes are stored in change tables or in an audit table. By default, changes are not stored.
For information about storing and applying changes, see *Working with Change Tables* (page 1283) and *Using an Audit table* (page 1289).

 When the **Bidirectional** replication profile is selected, the **Store Changes** button will be unavailable.
6. Click **OK** to close the **New Task** dialog box and save your settings.

Bidirectional replication

Bidirectional replication enables organizations to synchronize data between two endpoints (henceforth referred to as Endpoint A and Endpoint B), ensuring that both endpoints contain identical records. The endpoints can either be the same type (e.g. Oracle-to-Oracle) or different types (e.g. Microsoft SQL Server-to-Oracle). To implement bidirectional replication, two **Bidirectional Replication** tasks need to be defined: one

that captures changes made to Endpoint A and replicates them to Endpoint B (Task 1) and another that captures changes made to Endpoint B and replicates them to Endpoint A (Task 2). An explanation of how to set up these tasks is provided in the following sections.

Limitations

The following limitations apply to bidirectional replication tasks:

- Bidirectional replication does not currently support conflict resolution. To prevent conflicts, organizations should ensure that the application that updates the endpoints participating in a bidirectional replication task, does not simultaneously update the same record in both endpoints. In other words, if a record in Endpoint A was updated, the equivalent record in Endpoint B should only be updated after the update from Endpoint A is replicated to Endpoint B.
- Bidirectional replication tasks currently only support DDL statements from one of the sources involved in the task only. This means that for one of the sources, you must set the [Apply Changes DDL Handling Policy](#) to "Ignore" for all of the options (DROP, TRUNCATE, and ALTER).



The CREATE TABLE DDL is not supported.

- To ensure that the source and target endpoints are identical, transformations and filters should not be used in bidirectional replication tasks.
- The **Use direct path full load** option in the Oracle target endpoint settings is not supported.
- The **Stopping the Task after Full Load** options in the task settings' [Full Load Settings](#) tab is not supported.
- The task's **Change Processing Mode** must be set to [Transactional apply](#).

Supported endpoints

Bidirectional tasks support the following endpoints:

Source Endpoints:

- Amazon RDS for MySQL
- Amazon RDS for PostgreSQL
- Amazon RDS for SQL Server
- AWS Aurora Cloud for PostgreSQL
- File Channel
- Google Cloud SQL for MySQL
- Google Cloud SQL for PostgreSQL
- IBM DB2 for iSeries
- IBM DB2 for LUW
- IBM DB2 for z/OS
- Microsoft Azure SQL (MS-CDC)
- Microsoft SQL Server
- Microsoft SQL Server (MS-CDC)
- MySQL

- Oracle
- PostgreSQL
- SAP Sybase ASE

Target Endpoints:

- File Channel
- Google Cloud SQL for MySQL
- Google Cloud SQL for PostgreSQL
- IBM DB2 for z/OS
- Microsoft SQL Server
- Microsoft Azure SQL Database
- MySQL
- ODBC
- Oracle
- PostgreSQL
- SAP Sybase ASE

Setting up Bidirectional replication

This section describes how to set up a Bidirectional replication task in Qlik Replicate.

To set up Bidirectional Task 1:

1. Define a **Bidirectional Replication** task that replicates data from Endpoint A to Endpoint B.



*In a bidirectional replication task, **Full Load** replication is not enabled by default since it is assumed that both endpoints contain identical tables. If this is not the case (for instance, if Endpoint A contains tables that do not exist in Endpoint B), enable **Full Load** replication as well.*

2. Specify a source and target **Loopback prevention table schema** in the task settings' **Loopback Prevention** tab. For more information about loopback prevention settings, see *Bidirectional* (page 1171).
3. Run the task.

To set up Bidirectional Task 2:

1. Define another **Bidirectional Replication** task that replicates data from Endpoint B to Endpoint A.
2. Specify a source and target **Loopback prevention table schema** in the task settings' **Loopback Prevention** tab. For more information about loopback prevention settings, see *Bidirectional* (page 1171).
3. If Full Load was enabled when replicating data from Endpoint A to Endpoint B, you must first wait for the Full Load replication to complete before running the task. Otherwise, run the task immediately.

Using bidirectional replication with the File Channel endpoint

You can use bidirectional replication together with the File Channel endpoint. This is useful if you need to synchronize two endpoints that are either not able to communicate with each other (i.e. are not physically connected) or are located in the WAN. The process involves setting up six separate tasks: Two Full Load-only Unidirectional tasks and four Apply Changes-only Bidirectional tasks.

For information on setting up the File Channel endpoint, see *Using the Qlik Replicate file channel (page 1061)*.

To set up bidirectional replication with File Channel Endpoints:

1. Set up and run two Full Load only Unidirectional tasks.

Example (FC = File Channel):

Task 1: MySQL --> FC Target **Task 2:** FC Source --> Oracle

2. Wait for the Full Load-only tasks to finish.

3. Set up and run four Apply Changes-only Bidirectional tasks.

Example (FC = File Channel):

Task 1: MySQL Source --> FC Target **Task 2:** FC Source 1 --> Oracle Target

Task 3: Oracle Source --> FC Target 2 **Task 4:** FC Source 2 --> MySQL Target

Editing and viewing a task description

You can provide a description for specific tasks and then easily view or edit that description as required.

To provide, view or edit a task description:

1. Open the desired task.
2. In Designer or Monitor view, click the **Description** toolbar button.
3. Enter a description.
-OR-
Edit or view an existing description.
4. Click **OK**.

8.3 Defining and managing endpoints

Qlik Replicate requires information to connect to the source and target endpoints that you want to use in a task. For a list of endpoints you can work with in Qlik Replicate, see *Supported Platforms and Endpoints (page 1313)*.

You use the **Manage Endpoint Connections** window to add endpoints and edit and view the endpoint connection information.



The name cannot exceed 32 characters, contain non-Latin characters, or contain any of the following characters: | \ / : * ? " < >

- *Defining an endpoint (page 115)*
- *Editing endpoint configuration information (page 115)*
- *Viewing endpoint configuration information (page 116)*
- *Testing an endpoint connection (page 116)*
- *Duplicating endpoints (page 116)*
- *Searching for endpoints (page 117)*
- *Deleting endpoints (page 117)*

Defining an endpoint

Before you can begin to design a task, you must add endpoints to the Replicate server. To use an endpoint, you must have access to it somewhere in your system. When you add the endpoint to the Replicate server, you must provide connection information and proper user credentials.

Once you add endpoints to the Replicate server, you can begin to use them to build a replication task. For information on how to add an endpoint to a replication task, see *Adding a source and target endpoint to a task (page 117)*.

To add an endpoint:

1. In the Tasks view, click **Manage Endpoint Connections**.
The **Manage Endpoint Connections** window opens.
2. In the **Manage Endpoint Connections** window, click **New Endpoint**.
3. Select the type of endpoint you are using. The information that you must enter depends on which endpoint you select.
For more information, see the chapter that describes the endpoint you are using. For a list of supported databases, see *Supported Platforms and Endpoints (page 1313)*.

Editing endpoint configuration information

After you add the endpoint to the Replicate server and provide the connection information, you can make changes to some of the information.



You cannot change the following information in the endpoint window:

- *The name you provided for the endpoint.*
- *The endpoint **Type**, for example Oracle or Microsoft SQL Server.*
- *The endpoint **role**, either SOURCE or TARGET.*

To edit endpoint configuration information:

1. In the **Manage Endpoint Connections** window, select the endpoint you want to edit.
OR
In the **Endpoints** list on the left of the Designer view, double-click the endpoint you want to edit. Note that this option is only available when editing a specific task.
The **Manage Endpoint Connections** window opens with the selected endpoint settings.

2. Make changes to the information in any of the tabs in the window.
For more information, see the chapter for the specific Qlik Replicate endpoint you are using.
For information which endpoints are supported by Qlik Replicate, see *Supported Platforms and Endpoints* (page 1313).

Viewing endpoint configuration information

After you add the endpoint to the Replicate server and provide the connection information, you can view the information in the **Manage Endpoint Connections** window.

To view endpoint configuration information:

- Select an endpoint from the **Endpoints** list in the left pane; then click the tabs to view the information.

Testing an endpoint connection

You can try to contact the endpoint to make sure that you are connected to the endpoint you want to work with.

To test the endpoint connection:

1. In the **Manage Endpoint Connections** window, select the endpoint you want to work with.
2. At the bottom of the endpoint's **General** tab, click **Test Connection**.
If the connection is successful, a success message is displayed and a green check mark icon appears next to the **Test Connection** button.
If the connection fails, an error message is displayed at the bottom of the dialog box and the **View Log** button becomes available.
3. If the connection is successful, click **Close**.
If the connection fails, click **View Log** to view the server log entry with information for the connection failure.

Duplicating endpoints

You can duplicate an endpoint if you need to define a new endpoint with similar settings. Except for the name, all endpoint settings are duplicated to the new endpoint.

To duplicate an endpoint:

1. In the left panel of the **Manage Endpoint Connections** window, click the endpoint you want to duplicate.
2. Click **Duplicate**.
3. On the General tab, edit the name for the endpoint.
4. Make any other necessary changes.
5. Click **Save**; then click **Close**.

Searching for endpoints

You can search for endpoints by typing a sequence of letters in the **Filter by** box above the endpoints list. For example, to search for all endpoints whose names contain the string "Oracle", type "or". Only endpoints that match the search string are displayed.

Deleting endpoints

You can delete endpoints that you no longer require. Note that to delete an endpoint that is defined as a source or target in a task, you first need to remove the endpoint from the task.

To delete an endpoint:

- In the left panel of the **Manage Endpoint Connections** window, Select the endpoint and click **Delete**.

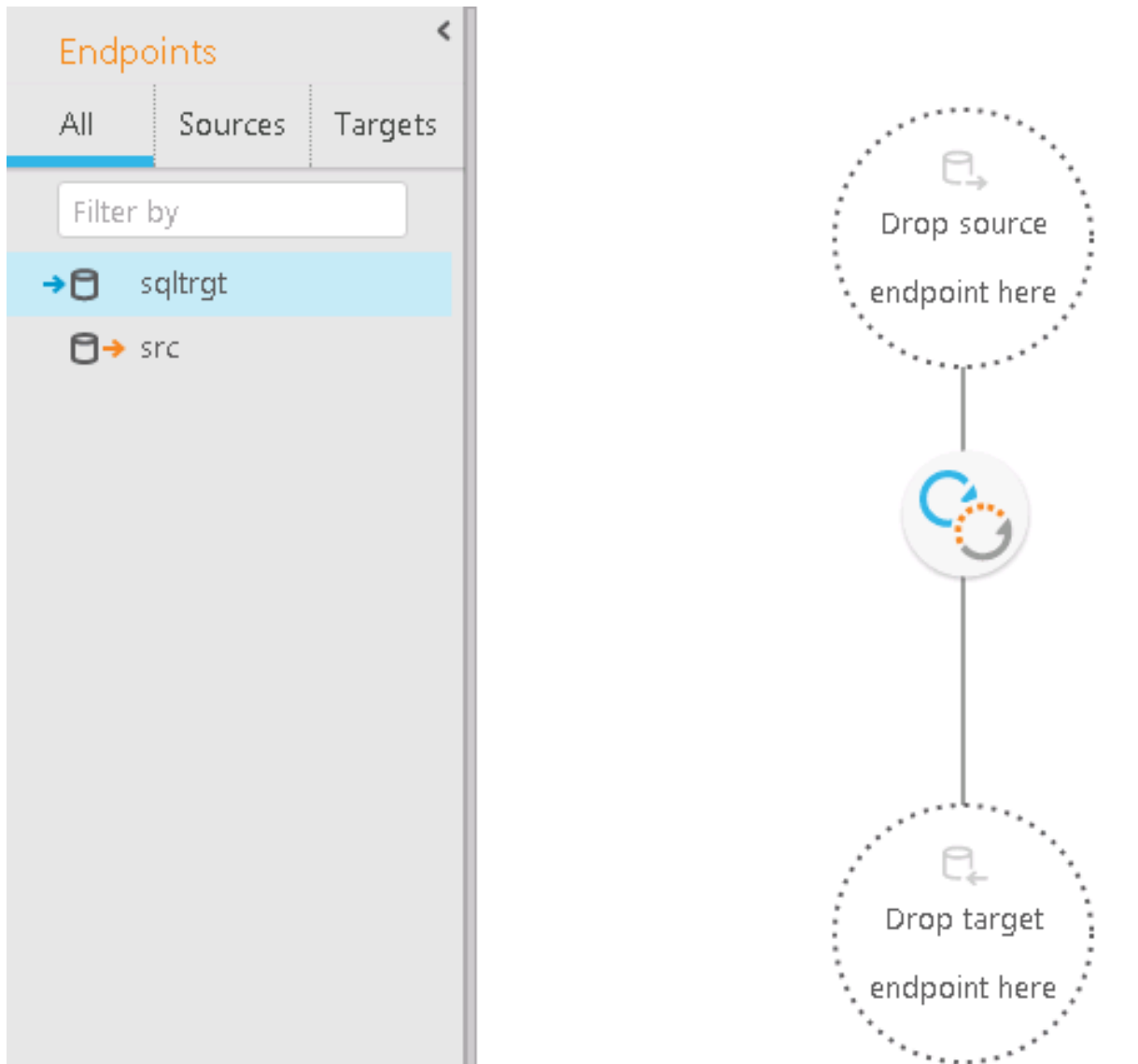
8.4 Adding a source and target endpoint to a task

Once you have added the endpoints, you can design the replication task. The first step in this process is to define the source endpoint where your data is currently stored and the target endpoints where you want to replicate the data. To do this, you just drag one of the endpoints you added into the task map (in Designer mode).

Once you select the endpoint for your task, you must select the tables from the source endpoint to be replicated. The next step in creating a replication task is *Selecting tables and/or views for replication* (page 118).

To add source and target endpoints to a task:

1. Do one of the following:
 - Create a new task. When you click **OK** in the **Create New Task** dialog box, the task opens on a dedicated tab. For more information, see *Adding tasks* (page 110).
 - In the Tasks view, select the task to which you want to add endpoints and click **Open**. The task opens on a dedicated tab.
2. The Task map is displayed, with the available endpoints listed in the pane on the left, as shown in the following figure.



3. Drag a source endpoint to the top circle in the task map (that contains the text **Drop source endpoint here**). If dragging is not possible, make sure that the endpoint you are using is defined as a source endpoint.
4. Drag a target endpoint to the bottom circle in the task map (that contains the text **Drop target endpoint here**). If dragging is not possible, make sure that the endpoint you are using is defined as a target endpoint.
5. Click **Save**.

8.5 Selecting tables and/or views for replication

This procedure describes how to select the source tables or views that you want to replicate. Note that tables can be selected from any supported endpoint, but views can only be selected from the following endpoints:

- Teradata
- Amazon Redshift

- PostgreSQL
- MySQL
- SAP Sybase ASE
- IBM DB2 for LUW
- IBM DB2 for z/OS
- Oracle
- Microsoft SQL Server
- ODBC with CDC
- ODBC



- *Replication of views is supported in Full Load Only tasks only, except when replicating from the following sources:*
 - *Amazon Redshift*
 - *Teradata*
 - *ODBC with CDC*
- *Views are replicated to the target endpoint as tables*
- *When replicating views, the corresponding tables are created without a primary key. This presents an issue for Apply Changes tasks, which require the target tables to have a primary key. Therefore, if you are also running Apply Changes tasks (using one of the CDC-capable endpoints mentioned above), you need to define one or more primary keys for each of the target tables using a transformation. For an explanation of how to accomplish this, see [Using the Transform tab \(page 1086\) in Defining transformations for a single table/view \(page 1084\)](#).*



When working with ODBC with CDC and Teradata source endpoints, any views and tables that you want to replicate must have the same context field(s). If you only want to replicate views, then all of the views must have the same context field(s).

For information on setting up context fields, see [Configuring Change Processing Settings](#).

For information on setting up context fields, see "Configuring Change Processing Settings" in the [Qlik Replicate Setup and User Guide](#).

Once you have selected tables/views to replicate, you can run the replication task. However, if you need to make any changes to the structure of the tables in the target endpoint or only select specific columns, you will need to carry out one or both of the following procedures:

- *Defining transformations for a single table/view (page 1084)*
- *Using filters (page 1092)*

To select tables/views:

1. Open the task you are working with if it is not already displayed in a dedicated tab.
For information on opening a task, see *Editing a replication task* (page 125).
2. In Designer mode, on the right side, click **Table Selection**.
If the source endpoint does not support view selection, the **Select Tables** dialog box opens. If the [source endpoint supports view selection](#), the **Select Tables/Views** dialog box opens.
See the following for information on how to work with the **Select Tables/Select Tables/Views** dialog box:
 - *Searching for tables/views to use in a replication task* (page 120)
 - *Selecting specific tables/views for replication* (page 121)
 - *Creating table/view selection patterns* (page 122)
 - *Setting load order* (page 124)

Searching for tables/views to use in a replication task

This topic walks you through searching for specific tables/views in preparation for including them in a replication task. You first search for tables that match specific criteria. Then you select the required tables/views from the search results to include them in the task. You can also carry out another search with new criteria and then add additional tables/views to the replication task.

After you finish searching, you can select tables/views for replication. Continue with *Selecting specific tables/views for replication* (page 121).

To search for tables/views to use in a replication task:

1. In **Designer** mode, click **Table Selection**.
2. In the **Select Tables** dialog box, if the source endpoint supports view selection, select one of the following:
 - **All** to search for tables and views
 - **Tables** to search for tables only
 - **Views** to search for views onlyOtherwise, skip to the next step.
3. From the **Schema** drop-down list, select a table/view schema.



When working with Microsoft SQL Server, quote characters in schema names - such as "\" or "[" - must be doubled in order to show the list of tables in the specified schema. For example, **My\Schema** should be specified as **My\\Schema**.



When selecting tables from the SAP Application or SAP Application (DB) endpoint, "Business Groups" will appear instead of "Schema".

4. Optionally, in the **Table/View** field, type the name or partial name of a table/view you wish to replicate. If you do not do this, Replicate will search for all of the tables/views in the selected schema.

i You can also include special characters in your search string. For more information, see the Note in *Creating a record selection condition for one or more columns* (page 1094).

- Click **Search** to display a list of tables/views.

i When selecting tables from the SAP Application or SAP Application (DB) endpoint, the **Table List** will display all of the tables in the selected Business Group. Hovering your mouse cursor over a table will display a tooltip as shown below.

Table List

FI - FINANCE DOCUM...	AGKO	
FI - FINANCE DOCUM...	BFI	
FI - FINANCE DOCUM...	BFI	

Source: AGKO
Class: Transparent
Description: Cleared Accounts

The **Table List** field displays any table/view that matches the specified search criteria.

If the source endpoint supports view selection, an additional **Type** column indicates whether the database object is a table or a view.

- Click **OK**.

See also: *Creating table/view selection patterns* (page 122).

Selecting specific tables/views for replication

This topic walks you through selecting specific tables/views to replicate.

When you select specific tables/views, all selected tables/views are replicated in full unless you define transformations or filters for the table/view. If you need to make changes to the table/view structures in the target endpoint or if you only want to select specific columns, then you need to perform the procedures described in *Defining transformations for a single table/view* (page 1084) and *Using filters* (page 1092) respectively.

To select specific tables/views:

- Open the **Select Tables/Views** dialog box.
- Select a **Schema**.
- Optionally, select the **Use exact table name** check box. This option is useful if your schema contains numerous tables as it will save you having to scroll through the entire list to find one specific table.
- If you selected the **Use exact table name** check box, type the exact name of the table you want to replicate in the **Table/View** field.
- Click **Search**.

The table or tables (If you did not select the **Use exact table name** check box) will be shown in the search results.

6. Select the table by adding it to the list on the right.
7. To add additional tables from the same schema, repeat steps 3-6. To add additional tables from a different schema, repeat steps 2-6.
8. Click **OK** to save your settings.



*If you rename a table in the database, the **Designer** tab will still show the original table name. The **Monitor** tab, on the other hand, will show the new table name.*

Removing specific tables/views from a replication task

This topic walks you through removing specific tables/views from the replication task.

To remove tables from the Selected Tables list:

1. From the **Selected Tables** list, select a table that you want to remove from the replication task and then click the button with a single left-facing arrowhead (**Remove**).
2. To remove all of the tables/views from the **Selected Tables** or **Selected Tables/Views** list, click the button with two left-facing arrowheads (**Remove All**).
3. Click **OK** to close the **Select Tables** or **Select Tables/Views** dialog box.
4. Click **Save** to make sure that Qlik Replicate saves the table information for this task.

Creating table/view selection patterns

This topic walks you through selecting tables/views using patterns. For example, you can include all tables/views that belong to the HR schema except for one or two tables/views that you exclude. You can also only exclude one or more table/view schemas or tables/views. This replicates the entire endpoint, except for those tables/views that you excluded.

The following example shows a pattern that replicates all tables that are members of the **dbo** schema except for the **dbo.PRODUCT_1%** table.

```
InclUde dbo.%  
ExcLude dbo.PRODUCT_1%
```

You can also use the "_" wildcard character to match a single character. For example, specifying `ExcLude m_d%` will exclude all tables that begin with `m` and end with `d%`, such as `model` or `msdb`.



Do not escape wildcard characters as this will instruct Replicate to interpret them as standard characters. As escape character conventions differ across databases, you should consult your database Help for guidance about supported escape characters. Some examples (where an underscore is the wildcard character) are as follows:

- *MySQL and PostgreSQL - \ _*
- *Microsoft SQL Server - [_]*
- *Oracle - For Oracle, use the `escapeCharacter` internal parameter to define a custom escape character.*

When you explicitly select tables/views, all selected tables/views are replicated in full unless you define transformations or filters for the table/view. If you need to make changes to the table/view structures in the target endpoint or if you only want to select specific columns, then you need to perform the procedures described in *Defining transformations for a single table/view* (page 1084) and *Using filters* (page 1092) respectively.



*To view all of the tables/views included when you use a table selection pattern, click the **Full Table List** tab in Designer view. The **Full Table List** lists all of the tables/views included in any table pattern you defined as well as all explicitly selected tables/views. To view only patterns and explicitly selected tables/views, click the **Patterns and Selected Tables** tab in Designer view.*

To create table/view selection patterns:

1. In the Designer view, in the **Select Tables/Views** dialog box, do any of the following:
 - Select a schema from the **Schema** drop-down list. All tables/views that belong to that schema are included in the table/view selection pattern.
 - Type the name or partial name of a table/view in the **Table/View** field. Any string that you enter here is included in the table/view selection pattern.
 - If the table/view that you type here is a member of the schema you selected in the **Schema** drop-down list, then you only have to type the name of the table/view.
 - If you did not select a schema or the table/view belongs to another schema, include the schema with the table name in the following format: `HR.Employees`, where HR is the schema.
2. Click **Include** to include all of the tables/views that match the selection criteria.
3. Click **Exclude** to exclude any tables that match the selection criteria.
4. Click **OK** to close the **Select Tables/Views** dialog box.
5. Click **Save** to make sure that Qlik Replicate saves the table/view information for this task.

Excluding specific tables from the replication task

You can easily exclude specific tables from being replicated.

To do this:

1. Open the the **Select Tables/Views** dialog box.
2. Select a **Schema** and then click **Search**.
Any tables in that schema will be shown in the search results.
3. Select the tables by adding them to the list on the right.
4. Click the **Include** button.
Include <schema_name>.% will be added to the **Table Selection Patterns** list.
5. Select the **Use exact table name** check box.
6. Type the name of the table you want to exclude in the **Table/View** field.
7. Click the **Exclude** button.
Exclude <schema_name>.<table_name> will be added to the **Table Selection Patterns** list.
8. To exclude additional tables from the same schema, repeat Steps 6-7. To exclude tables from a different schema, clear the **Use exact table name** check box and then repeat Steps 2-7.
9. Click **OK** to save your settings.



Filters containing wildcard escape characters that excluded/included tables during Full Load will not exclude/include matching tables added during CDC. For example, if there is an exclude pattern `dbo.pc[_]%` and a new table `dbo.pc_table2` is created during CDC, the table will be added to replication task (as opposed to being excluded).

Setting load order

You can set the load order for each of the selected tables. This may be useful, for example, if your selected tables list contains tables of different sizes and you want the smaller tables to be loaded before the larger tables. When a group of tables are set with the same load order, Replicate will load the tables according to the table ID.

Load order can be set and modified (see [note](#) below) in the following places:

- The **Select Tables [window](#)** (opened in Designer view by clicking the **Table Selection** button in the right of the console).
- The **Patterns and Selected Tables** list in the right of the console (in Designer view).



- *Load order cannot be changed while the task is running. If you want to change the load order, first stop the task, then change the load order as desired, and finally reload the target.*
- *Load order cannot be set for "Exclude" patterns.*

To set the load order for a specific table:

1. Select the desired table in the **Selected Tables** list.
2. From the **Load Order** drop-down list, select one of the available priority levels (Lowest Priority, Low Priority, Normal Priority, High Priority, and Highest Priority).
3. This step is only relevant if you are setting load order in the **Select Tables** window. Click **OK** to save your settings and close the **Select Tables** window.

To set the same load order for multiple tables:

1. Select the desired tables in the **Selected Tables** list.
2. From any of the selected items' **Load Order** drop-down list, select one of the available priority levels.
3. This step is only relevant if you are setting load order in the **Select Tables** window. Click **OK** to save your settings and close the **Select Tables** window.

8.6 Editing a replication task

You can make changes to tasks that you previously created. Just open the task and make the changes in the same way that you did when you created the task.

To edit a task:

1. In **Tasks** view, select the task and click **Open**.
The task opens, displaying the source and target endpoints and which tables have been selected for replication.
2. Continue with any of the following procedures:
 - *Adding a source and target endpoint to a task (page 117)*
 - *Selecting tables and/or views for replication (page 118)*
 - *Defining transformations for a single table/view (page 1084)*
 - *Using filters (page 1092)*
 - *Task Settings (page 1165)*

8.7 Deleting a replication task

You can delete tasks that you created. To prevent complications, it is recommended not to use the name of a deleted task for a new task you create. Such a task would be created with the same settings as the deleted task.



If you use a Microsoft SQL Server endpoint, a Microsoft SQL Server system administrator must delete the Microsoft SQL Server Replication Publisher definitions for the endpoint that was used in the task from SQL Server.

*For more information, see the *Limitation and considerations (page 940)* in the *Microsoft SQL Server* chapter.*

To delete a task:

1. Stop the task that you want to delete.
2. In **Tasks** view, click **Delete**.
The task is deleted.

8.8 Searching for tasks

In **Tasks** view, you can search for tasks by typing a sequence of letters in the **Filter Tasks** box above the tasks. For example, to search for all tasks with names that begin with "Oracle-to", type "or". Only tasks that match the search string are displayed.

8.9 Migrating tasks

You can export replication tasks to a file. When exporting a task using the command line, all exported tasks are saved to the **imports** folder under <product_dir>/Qlik/Replicate/Data. When exporting a task using the Qlik Replicate Console, one of the following occurs (according to your browser settings):

- The task JSON file will be automatically downloaded to the default download location
- You will be prompted for a download location

You can import an export file (*.json) to another instance of the Qlik Replicate Server. This lets you use a task that you created in Qlik Replicate in a different environment. For example, if you created tasks in a development environment and now want to use the task in a production environment.

Importing and exporting a task can be done using either the command line or the Qlik Replicate Console. When exporting or importing a task using the command line, you must perform this task on the computer where Qlik Replicate is installed.



If you need to access the computer with Qlik Replicate from a remote computer, you can use a telnet connection.

When you export a task to a different environment, you may need to edit the task information. For example, you may need to change the connection string for an endpoint.

- *Exporting tasks (page 126)*
- *Editing an exported (json) file (page 129)*

Exporting tasks

The following section explains how to export a task using either the Qlik Replicate Console or the command line.

To export a task using the Qlik Replicate Console:

1. Switch to **Tasks** view (make sure you're in **Designer** mode).
2. Do one of the following:
 - In **TASKS** tab, select the task you want to export and then either click the **Export** toolbar button or right-click the task and select **Export**.
 - OR
 - In the **TASK_NAME** tab (opened when a task is being edited), click the **Export Task** toolbar button.

Depending on your browser settings, one of the following will occur:

- The task JSON file will be automatically downloaded to the default download location
- You will be prompted for a download location. In this case, save the JSON file to your preferred location.

To export a task using the command line:

1. From the Qlik Replicate computer where you defined the task you want to import, open the Qlik Replicate command line console by doing the following:
On Windows: Go to **All Programs** in the **Start** menu and point to **Qlik Replicate**, then to **Utilities** and then select **Qlik Replicate Command Line**.
A command-line console is displayed with the correct prompt for Qlik Replicate.



You can also open the Windows command-line console and change the directory to the following:

```
<Qlik Replicate Installation Directory>\bin
```

For example, to use the path to the folder or directory where Qlik Replicate is installed by default, type: `C:\Program Files\Attunity\Replicate\bin`.

On Linux: Run the following command in the `Replicate\bin` directory:

```
source ./arep_login.sh
```

2. At the prompt in the command-line console, type the following:
`repctl -d data-directory exportrepository task=task_name [folder_name=path]`
By default, a file called `<task_name>.json` containing the exported task settings is created in the `<product_dir>\data\imports` folder. If you want the file to be created in a different folder, include the `folder_name=path` parameter in the command.
After the file has been created, you can import it into another Qlik Replicate instance as described in *Importing tasks* (page 127).



If the `<product_dir>\data` folder was installed in a non-default location during the installation - OR - if it was later moved to a non-default location, you need to tell Replicate where the folder is located.

This is done by including the `-d <data_folder>` parameter in the command.

Example:

```
repctl -d D:\Data exportrepository task=mytask
```

Importing tasks

The following section explains how to import a task using either the Qlik Replicate Console or the command line.



Before importing tasks, make sure that the tasks on the destination server are either running or stopped (i.e. not in the process of being started).

To import a task using the Qlik Replicate Console:

1. Switch to **Tasks** view (make sure you're in **Designer** mode).
2. Click the **Import Task** toolbar button.
The **Import Task** dialog box opens.
3. Browse to the task JSON file you want to import and then click **Import Task**.
The task is imported.

To import a task using the command line:

1. From the Qlik Replicate computer where you want to import the task, open the Qlik Replicate command line console by doing the following:
From the **Start** menu, expand **Qlik Replicate** and then select **Qlik Replicate Command Line**.
A command-line console is displayed with the correct prompt for Qlik Replicate.



You can also open the Windows command-line console and change the directory to the following:

<product_dir>\Attunity Replicate>\bin

For example to use the path to the directory where Qlik Replicate is installed by default, type:

C:\Program Files\Attunity\Replicate\bin.

2. Run the following command:
`repctl connect`
3. To import the task, run the following command:
`importrepository [-d data-directory] json_file=<Full path to the exported *.json file>`

Example:

```
importrepository -d D:\MyData json_file=C:\Temp\many_tables.json
```

The exported *.json file will be located in the <data-directory>\imports folder on the original computer where the task was exported or in the folder specified by the folder_name parameter in the export command.

Example:

```
<product_dir>\data\imports\many_tables.json
```



If the <product_dir>\data folder was installed in a non-default location during the installation - OR - if it was later moved to a non-default location, you need to tell Replicate where the folder is located.

This is done by including the -d data_folder parameter in the command.

Example:

```
repctl -d D:\MyData importrepository json_file=C:\mytask.json
```

If you are importing this task into a different environment, you should copy the file to a location on the second Qlik Replicate computer and then import the file from there.

In many cases, when you import the task into a different environment, you will need to make changes to the task. For example, you may need to change the connect strings for the endpoints in the task or change the user password. In that case, you will need to edit the *.json file.

See *Editing an exported (json) file* (page 129) for more information.

Editing an exported (json) file

You can open the *.json file in any plain text editor. It is possible to make changes to any of the sections in the file; however, be sure that you only change the data and not the field names. For example, the entry "name": "DB_Name" displays the name field for a source table in a defined endpoint. In this case, you can change the data "DB_Name" but not the included metadata ("name").



*Make any changes to the *.json file before you carry out the import operation.*



Information about the endpoints, tables, tasks, task settings, and logger settings should be changed using the Qlik Replicate Console after the file is imported.

*To be able to use the new task, you will need to make changes to the endpoint password and connection strings by editing the *.json file. See *Making changes to the endpoint connection information* (page 129) for more information.*

Making changes to the endpoint connection information

In the "endpoints" section, you can make changes to the connection string and the password. The following is an example of the "endpoints" section of the *.json file.

```
"endpoints": [{
  "name": "Oracle_Source",
  "type": "Oracle",
  "connection_string": "server= bee01-xp:1521/xe;username=SYSTEM",
  "authenticator": "
{01000000D08C9DDF0115D1118C7A00C04FC297EB010000003EA495B32CAAE14CB9777B96B3CC00B3000000002000
00000003660000A8000000100000002765A3287AB56447DA31508F71CE6270000000004800000A00000001000000
088D5C1BBD615BEEAF5FAC1B9B0E20800800000075D89177A9C6F11B140000047B3110B80920DD9EB0A5FABA0567
9979B78DDD0}",
  "role": "SOURCE"
}, {
  "name": "SQLSERVER_Target",
  "type": "SQLServer",
  "connection_string": "server=bee01-
xp;endpoint=tempdb;windowsAuthentication=Y;CDCBCP=Y;FullloadBCP=Y;BCPPacketSize=16384",
  "role": "TARGET"
}
```

To change the connect string:

1. In the *.json file, under "endpoints", find "connection string".
For example, "connection_string": "server= bee01:1521/xe;username=SYSTEM".
2. Change the information after the equal signs (=) as necessary.

For example, if the endpoint in the new environment is on a computer called B2, change `server=bee01` to `server=B2`.



Make sure that the connect string remains between the quotation marks ("").

To change an endpoint password:

1. In the *.json file, under "endpoints", find "authenticator".

For example, "authenticator": "

```
{01000000D08C9DDF0115D1118C7A00C04FC297EB010000003EA495B32CAAE14CB9777B96B3CC00B3000000  
000200000000003660000A8000000100000002765A3287AB56447DA31508F71CE6270000000004800000A  
00000001000000088D5C1BBD615BEEAF5FAC1B9B0E20800800000075D89177A9C6F11B1400000047B3110B  
80920DD9EB0A5FABA05679979B78DDD0}
```



The password is presented as an encrypted string.

2. Change the password string to the relevant password for the endpoint in the new environment. Type the new password using plain text exactly as it is configured in the endpoint. For example, `8yTkLmT`. When you save the file and then import it to the new environment, the password is encrypted automatically.

9 Adding and managing source endpoints

This topic describes how to configure source endpoint settings. When you define a replication task, the source endpoint is the data repository containing the data that you wish to replicate.

In this section:

- *Using AWS Aurora Cloud for PostgreSQL as a source (page 132)*
- *Using Amazon RDS for SQL server as a source (page 146)*
- *Using Amazon RDS for MySQL as a source (page 160)*
- *Using Amazon RDS for PostgreSQL as a source (page 170)*
- *Using ARC CDC solutions in Qlik Replicate (page 185)*
- *Using a file as a source (page 195)*
- *Using Google Cloud AlloyDB for PostgreSQL as a source (page 211)*
- *Using Google Cloud SQL for MySQL as a source (page 224)*
- *Using Google Cloud SQL for PostgreSQL as a source (page 234)*
- *Using Hadoop as a source (page 247)*
- *Using IBM DB2 for iSeries as a source (page 256)*
- *Using IBM DB2 for LUW as a source (page 267)*
- *Using IBM DB2 for z/OS as a source (page 276)*
- *Using IBM Informix as a source (page 306)*
- *Using Microsoft Azure SQL Managed Instance as a source (page 312)*
- *Using Microsoft Azure SQL (MS-CDC) as a source (page 325)*
- *Using Microsoft SQL Server (MS-CDC) as a source (page 337)*
- *Using Microsoft SQL Server as a source (page 348)*
- *Using Microsoft Azure Database for MySQL as a source (page 368)*
- *Using a MongoDB database as a source (page 378)*
- *Using a MySQL-based database as a source (page 385)*
- *Using ODBC to connect to a source (page 397)*
- *Using ODBC with CDC as a source (page 405)*
- *Using Oracle as a source (page 416)*
- *Using PostgreSQL as a source (page 452)*
- *Using Salesforce (Streaming CDC) as a source (page 467)*
- *Using Salesforce (Incremental Load) as a source (page 475)*
- *Using SAP Application as a source (page 483)*
- *Using SAP Application (DB) as a source (page 498)*
- *Using SAP Extractor as a source (page 511)*
- *Using SAP HANA as a source (page 537)*
- *Using SAP ODP as a source (page 549)*
- *Using SAP Sybase ASE as a source (page 562)*
- *Using Teradata Database as a source (page 570)*

9.1 Using AWS Aurora Cloud for PostgreSQL as a source

This section describes how to set up and use an AWS Aurora Cloud for PostgreSQL database as a source in a replication task.

In this section:

- *Prerequisites (page 133)*
- *Required permissions (page 133)*
- *Using an account without the "superuser" role (page 134)*
- *Setting up AWS Aurora Cloud for PostgreSQL for CDC (Change data capture) (page 135)*
- *Limitations and considerations (page 136)*
- *Supported source data types (page 137)*
- *Homogeneous Replication (page 140)*
- *Setting general connection properties (page 142)*
- *Setting advanced connection properties (page 144)*
- *Removing Replicate artifacts from the AWS Aurora Cloud for PostgreSQL database (page 145)*

Prerequisites

The following section describes the prerequisites for working with Qlik Replicate and an AWS Aurora Cloud for PostgreSQL database source.

Download PostgreSQL 13 ODBC driver, and client programs and libraries.

- **Example PostgreSQL ODBC file name:** postgresql13-odbc-13.02.0000-1PGDG.rhel8.x86_64.rpm
- **PostgreSQL client programs and libraries file name:** postgresql13-13.8-1PGDG.rhel<version>.x86_64.rpm

Then, open a shell prompt and do the following:

1. Stop the Replicate service:

```
./areplicate stop
```
 2. Optionally, confirm that the service has stopped:

```
./areplicate status
```
 3. Install PostgreSQL 13 client programs and libraries.
 4. Install the ODBC driver for PostgreSQL 13.
 5. Change the working directory to `<Replicate-Install-Dir>/bin`.
 6. Copy the driver location to the `site_arep_login.sh` file:

```
echo "export LD_LIBRARY_PATH=\$LD_LIBRARY_PATH:/usr/lib:/usr/lib64:/usr/pgsql-13/lib" >> site_arep_login.sh
```

This will add the driver to "LD_LIBRARY_PATH" and update the driver location in the **site_arep_login.sh** file.
 7. Optionally, confirm that the driver location was copied:

```
cat site_arep_login.sh
```
 8. Makes sure that the `/etc/odbcinst.ini` file contains an entry for PostgreSQL, as in the following example:

```
[PostgreSQL Unicode(x64)]
Description = PostgreSQL ODBC driver
Driver = /usr/pgsql-13/lib/psqlodbc.so
Setup = /usr/pgsql-13/lib/psqlodbcw.so
UsageCount = 1
```
 9. Start the Replicate service:

```
./areplicate start
```
 10. Optionally, confirm that the service has started:

```
./areplicate status
```
- When the **Apply Changes** task option is enabled, the user specified in the PostgreSQL source database's **General** tab must be granted super-user permissions.

Required permissions

The user specified in the **General** tab when *Setting general connection properties* (page 142) must be granted the following permissions in the AWS Aurora Cloud for PostgreSQL database:

- **For Full Load replication:** Standard SELECT on the source database
- **For Apply Changes replication:** See *Setting up AWS Aurora Cloud for PostgreSQL for CDC (Change data capture)* (page 135).

- When the **Apply Changes** and/or **Store Changes** task replication option is enabled, the user specified in the AWS Aurora Cloud for PostgreSQL source endpoint's **General** tab must be granted superuser permissions.

Using an account without the "superuser" role

If you are not using an account with the **rds_superuser** role, you need to create several objects to capture Data Definition Language (DDL) events. Create these objects in the selected account and then create a trigger in the master user account.

To do this:

1. Choose a schema where you want the objects to be created. The default schema is `public`. Ensure that the schema exists and is accessible by the non-privileged account.

2. Create the table **attrep_ddl_audit** by running the following command:

```
create table <objects_schema>.attrep_ddl_audit
(
  c_key      bigserial primary key,
  c_time     timestamp,      -- Informational
  c_user     varchar(64),    -- Informational: current_user
  c_txn      varchar(16),    -- Informational: current transaction
  c_tag      varchar(24),    -- Either 'CREATE TABLE' or 'ALTER TABLE' or
  'DROP TABLE'
  c_oid      integer,        -- For future use - TG_OBJECTID
  c_name     varchar(64),    -- For future use - TG_OBJECTNAME
  c_schema   varchar(64),    -- For future use - TG_SCHEMANAME. For now,
  holds the current_schema
  c_ddlqry   text            -- The DDL query associated with the current
  DDL event
);
```

3. Create the function **attrep_intercept_ddl** by running the following command:

```
CREATE OR REPLACE FUNCTION <objects_schema>.attrep_intercept_ddl()
  RETURNS event_trigger
LANGUAGE plpgsql
  AS $$
  declare _qry text;
BEGIN
  if (tg_tag='CREATE TABLE' or tg_tag='ALTER TABLE' or tg_tag='DROP
  TABLE') then
    SELECT current_query() into _qry;
    insert into <objects_schema>.attrep_ddl_audit
    values
    (
      default,current_timestamp,current_user,cast(TXID_CURRENT() as
```

```
varchar(16)),tg_tag,0, '',current_schema,_qry
    );
    delete from <objects_schema>.attrep_ddl_audit;
end if;
END;
$$;
```

4. If you are logged in with a non-privileged account, log out of the non-privileged account and log in with an account that has the **rds_superuser** role assigned to it.



If the **attrep_intercept_ddl** stored procedure is not being created in the default schema, you need to specify the schema name in the **Create DDL artifacts in schema** field in the endpoint settings' **Advanced** tab.

For more information on the replication configuration parameters, see the PostgreSQL Help.

5. Create the event trigger **attrep_intercept_ddl** by running the following command:

```
CREATE EVENT TRIGGER attrep_intercept_ddl ON ddl_command_end
EXECUTE PROCEDURE <objects_schema>.attrep_intercept_ddl();
```

6. Grant the following permissions to the non-privileged account:

- GRANT INSERT ON attrep_ddl_audit to <non-privileged-user>;
- GRANT DELETE ON attrep_ddl_audit to <non-privileged-user>;
- GRANT USAGE ON attrep_ddl_audit_c_key_seq TO <non-privileged-user>;
- GRANT rds_replication to <non-privileged-user>;

7. Grant SELECT permission to the user:

```
GRANT SELECT ON ALL TABLES IN SCHEMA <SCHEMA_NAME> TO <non-privileged-user>;
```

8. Grant usage on the schema to the user:

```
GRANT USAGE ON SCHEMA <SCHEMA_NAME> TO <non-privileged-user>;
```

Setting up AWS Aurora Cloud for PostgreSQL for CDC (Change data capture)

You can use the AWS master user account for the PostgreSQL database instance as the user for the AWS Aurora Cloud for PostgreSQL source endpoint. The master user account has the required roles that allow it to set up change data capture (CDC).

If you prefer not to use the master user account, the account you wish to use must have the **rds_superuser** role and the **rds_replication** role. The **rds_replication role** grants permissions to manage logical slots and to stream data using logical slots.

To enable logical replication for an AWS Aurora Cloud PostgreSQL database instance:

1. Create a new DB cluster parameter group with the following configuration:
 - a. Set the **rds.logical_replication** parameter in the database parameter group to **1**.
 - b. **max_wal_senders** – The number of tasks that can run in parallel.
 - c. **max_replication_slots** – Every connection (i.e. task) creates a new slot; this parameter configures the amount of simultaneous connections that can be established to the database.
2. Link the AWS Aurora Cloud cluster DB to the DB cluster parameter group.



If the **attrep_intercept_ddl** stored procedure is not being created in the default schema, you need to specify the schema name in the [Create DDL artifacts under schema](#) field.

Limitations and considerations

The following limitations apply when AWS Aurora Cloud for PostgreSQL as a source:

- The database name cannot include a semi-colon (;).
- Both the source table and the corresponding target table must have an identical Primary Key. In the event that one of the tables does not have a Primary Key, the result of DELETE and UPDATE record operations will be unpredictable.
- The “Start Process Changes from Timestamp” run option is not supported.
- Replication of the Before Image is not supported.
- Replication of multiple tables with the same name but a different case (e.g. table1, TABLE1 and Table1) may cause unpredictable behavior and is therefore not supported.
- Change processing of [CREATE | ALTER | DROP] table DDLs are supported unless they are held in an inner function/procedure body block or in other nested constructs.

For example, the following change will not be captured:

```
CREATE OR REPLACE FUNCTION attu.create_distributors1() RETURNS void
LANGUAGE plpgsql
AS $$
BEGIN
create table attu.distributors1(did serial PRIMARY KEY,name varchar(40) NOT NULL);
END;
$$;
```

- Change processing of TRUNCATE operations is not supported.
- Deferred Constraints are not supported.
- **Partitioned tables**
 - CDC is not supported.
- When using the *Parallel Load (page 1098)* feature, table segmentation according to partitions or sub-partitions is not supported.
- UPDATES to a Unique Index segment will not be applied to the target, and a record of the attempted UPDATE(s) will be written to the attrep_apply_exceptions Control table.
- When using [filters](#) that do not contain a Primary Key, DELETE operations will not be captured.

- A DDL operation that occurs during Full Load, but that was not preceded by a DML operation will not be captured.
- PostGIS datatypes are not supported.
- Unique Indexes with CASE expressions are not supported.

Supported source data types

The following table shows the AWS Aurora Cloud for PostgreSQL target data types that are supported when using Qlik Replicate and the default mapping to the Qlik Replicate data types.




For information about data type mapping in homogeneous replication, see *Homogeneous Replication (page 140)*.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.


PostgreSQL and Qlik Replicate data types


AWS Aurora Cloud for PostgreSQL data types	Qlik Replicatedata types
INTEGER	INT4
SMALLINT	INT2
BIGINT	INT8
NUMERIC	NUMERIC(28,6)
NUMERIC(p,s)	If precision is =< 38, then: NUMERIC(p,s) If precision is => 39, then: STRING
DECIMAL	NUMERIC(28,6)
DECIMAL(p,s)	If precision is =< 38, then: NUMERIC(p,s) If precision is => 39, then: STRING
REAL	REAL4
DOUBLE	REAL8
SMALLSERIAL	INT2
SERIAL	INT4
BIGSERIAL	INT8

9 Adding and managing source endpoints

AWS Aurora Cloud for PostgreSQL data types	Qlik Replicatedata types
MONEY	NUMERIC(38,4) <div style="border: 1px solid gray; padding: 5px;">  <i>The MONEY data type is mapped to FLOAT in Microsoft SQL Server.</i> </div>
CHAR	WSTRING (1)
CHAR(n)	WSTRING (n)
VARCHAR(n) <div style="border: 1px solid gray; padding: 5px;">  <i>VARCHAR without a length (n) is not recognized as a valid data type by target endpoints. Consequently, if a source column data type is set to VARCHAR without an explicit length, Replicate will set a default length of 8000 bytes.</i> <i>You can change the default by setting the following internal parameter to the required length:</i> <code>unboundedVarcharMaxSize</code> <i>See also Internal Parameters.</i> </div>	WSTRING (n)
TEXT	NCLOB
BYTEA	BLOB
TIMESTAMP	DATETIME
TIMESTAMP (z)	DATETIME
	<div style="border: 1px solid gray; padding: 5px;">  <i>Replicate only supports ISO formatted textual DATE formats (the default). If other formats are used, an error will be generated. You can change the date format in the AWS Aurora Cloud for PostgreSQL.conf file or using the <code>PGDATESTYLE</code> environment variable. You can also change the date format at database level.</i> </div>
DATE	DATE
TIME	TIME
TIME (z)	TIME

AWS Aurora Cloud for PostgreSQL data types	Qlik Replicatedata types
INTERVAL	STRING (128) - 1 YEAR, 2 MONTHS, 3 DAYS, 4 HOURS, 5 MINUTES, 6 SECONDS
BOOLEAN	STRING (5)
ENUM	STRING (64)
CIDR	STRING (50)
INET	STRING (50)
MACADDR	STRING (18)
BIT (n)	STRING (n)
BIT VARYING (n)	STRING (n)
UUID	STRING
TSVECTOR	CLOB
TSQUERY	CLOB
XML	CLOB
POINT	STRING (255) "(x,y)"
LINE	STRING (255) "(x,y,z)"
LSEG	STRING (255) "((x1,y1), (x2,y2))"
BOX	STRING (255) "((x1,y1), (x2,y2))"
PATH	CLOB "(x1,y1),(xn,yn)"
POLYGON	CLOB "(x1,y1),(xn,yn)"
CIRCLE	STRING (255) "(x,y),r"
JSON	NCLOB

 *Reading and writing JSON data is performed using wide string (where each character occupies 2 bytes). Therefore, when replicating JSON columns to a PostgreSQL target, set the limited lob size to twice the maximum JSON length. So, for example, if the JSON data is 4 KB, set the limited lob size value to 8 KB.*

AWS Aurora Cloud for PostgreSQL data types	Qlik Replicatedata types
JSONB <div style="border: 1px solid gray; padding: 5px; margin: 5px 0;">  <p><i>Reading and writing JSONB data is performed using wide string (where each character occupies 2 bytes). Therefore, when replicating JSONB columns to a PostgreSQL target, set the limited lob size to twice the maximum JSONB length. So, for example, if the JSONB data is 4 KB, set the limited lob size value to 8 KB.</i></p> </div>	NCLOB
ARRAY	NCLOB
COMPOSITE	NCLOB
INT4RANGE	STRING (255)
INT8RANGE	STRING (255)
NUMRANGE	STRING (255)
TSRANGE	STRING (255)
INT4MULTIRANGE	STRING (255)
INT8MULTIRANGE	STRING (255)
NUMMULTIRANGE	STRING (255)
TSMULTIRANGE	STRING (255)
CHARACTER VARYING	If length is specified: WSTRING (LENGTH) If no length is specified: WSTRING (8000)
TINTERVAL	WSTRING(255)



The PostGIS data type is not supported.

Homogeneous Replication

The following section describes how Replicate handles replication from AWS Aurora Cloud for PostgreSQL to the following targets: PostgreSQL, Microsoft Azure Database for PostgreSQL, Google Cloud SQL for PostgreSQL, and Google Cloud AlloyDB for PostgreSQL.

In homogeneous replication, source column and table collations will be replicated to the target as described in *Column and table collation* (page 142).



In homogeneous replication, the source data first passes through the Qlik Replicate data type and is therefore subject to any limitations of that type.

For information on Replicate data types and their limitations (where relevant), see Replicate data types (page 81).

For information on which Replicate data types the source data passes through when replicating from AWS Aurora Cloud for PostgreSQL, see the AWS Aurora Cloud for PostgreSQL to Qlik Replicate data types mapping table described earlier.

Limitations and considerations

Homogeneous replication is not supported with the following Replicate configurations or operations:

- Apply Changes only tasks
- Stopping and resuming a task
- Starting a task from a timestamp
- When the **If target table already exists** in the task settings' **Full Load Settings** tab is set to **Do nothing**

Data type considerations and exceptions

When replicating from AWS Aurora Cloud for PostgreSQL database to a PostgreSQL-based target (for example, Microsoft Azure Database for PostgreSQL), source and target data types are identical, with the following exceptions:

Data type differences in homogeneous replication

AWS Aurora Cloud for PostgreSQL source	PostgreSQL-based target
ENUM	STRING
COMPOSITE	STRING
NUMERIC	NUMERIC (28,6)
NUMERIC (p,s) A numeric array with precision and scale.	NUMERIC A numeric array without precision and scale.
TINTERVAL	<div data-bbox="475 1720 544 1787" data-label="Image"> </div> <p><i>As PostgreSQL have discontinued support for the TINTERVAL data type, tables with this data type will not be created on the target if the target database version does not support it.</i></p>
INT4MULTIRANGE	STRING (255)

AWS Aurora Cloud for PostgreSQL source	PostgreSQL-based target
INT8MULTIRANGE	STRING (255)
NUMMULTIRANGE	STRING (255)
TSMULTIRANGE	STRING (255)

Column and table collation

When replicating from AWS Aurora Cloud for PostgreSQL database to a PostgreSQL-based target, column and table collations will be replicated to the target.



To support collation replication, the DBA must ensure that the collations defined for the source AWS Aurora Cloud for PostgreSQL database are the same as those defined for the PostgreSQL-based target database.

Non-nullable columns and primary/unique index names

Non-nullable columns and Primary/Unique Index names are preserved during homogeneous replication.

Replication of Partitioned Tables

In order to replicate partitioned tables from a AWS Aurora Cloud for PostgreSQL source to a PostgreSQL-based target, you need to do the following:


1. Manually create the parent and child tables in the target database.
2. Define a separate task to replicate to those tables.
3. Set the **Truncate before loading** option in the *Full Load Settings* (page 1171).

Setting general connection properties


This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add an AWS Aurora Cloud for PostgreSQL endpoint source database to Qlik Replicate:


1. In the Qlik Replicate console, click **Manage Endpoint Connections** to open the **Manage Endpoint Connections** dialog box. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints* (page 114).
2. In the **Name** field, type a name for your AWS Aurora Cloud for PostgreSQL database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the AWS Aurora Cloud for PostgreSQL database. This is optional.
4. Select **Source** as the database **role**.
5. Select **AWS Aurora Cloud for PostgreSQL** as the database **Type**.

 When this endpoint is used as a duplicated source in a Log Stream Staging setup, select the **Read changes from log stream** check box and then select the relevant Log Stream Staging task from the drop-down list.
For information on setting up and managing Log Stream Staging tasks, see *Using the Log Stream* (page 1072).

6. In the **Server** name field, specify the host name of the database instance, for example:
`myinstance.123456789012.us-east-1.rds.amazonaws.com`
7. Optionally, change the default **Port** (5432).
8. Enter the AWS Aurora Cloud for PostgreSQL database authentication information (**Username**, **Password**) of an authorized AWS Aurora Cloud for PostgreSQL user. If you do not know this information, see your AWS Aurora Cloud for PostgreSQL database system manager.

 Consider the following:

- This information is case sensitive.
- To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**. If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box. To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

 Make sure that the AWS Aurora Cloud for PostgreSQL database user entered in the AWS Aurora Cloud for PostgreSQL database Authentication section has the correct access privileges.

9. Type the **Database** name or select one from the list of available endpoints. This is the name of the AWS Aurora Cloud for PostgreSQL database from which you are replicating data.

SSL authentication options

In the **Security** section, you can configure Replicate to connect to the Amazon RDS for PostgreSQL database using SSL.

- **SSL Mode:** Select one of the following:
 - **Disable** - Connect with a surname and password only.
 - **Allow** - Establish an encrypted connection if requested by the server.
 - **Prefer** - Establishes an encrypted connection if the server supports encrypted connections, falling back to an unencrypted connection if an encrypted connection cannot be established.
 - **Require** - Establishes an encrypted connection if the server supports encrypted connections. The connection attempt fails if an encrypted connection cannot be established.

- **Verify CA** - Similar to **Required**, but also verifies the server Certificate Authority (CA) certificate against the configured CA certificates. The connection attempt fails if no valid matching CA certificates are found.
- **Verify Full** - Similar to **Verify CA**, but also performs host name identity verification by checking the host name the client (i.e. Replicate) uses for connecting to the server against the identity in the certificate that the server sends to the client. The client checks whether the host name that it uses for connecting matches the Common Name value in the server certificate. The connection fails if there is a mismatch.
- **Client certificate path** - The path to the client certificate requested by the server.
- **Secret key path** - The path to the client private key file in PEM format.
- **CA path** - The path to the Certificate Authority (CA) that issued the client certificate file in PEM format.
- **CRL path** - The path to the CRL certificate. This file contains certificates revoked by certificate authorities. If the server certificate appears in this list, the connection will fail.
- **SSL compression** - Select this option to compress the data before it is encrypted.

Setting advanced connection properties

In the **Advanced** tab, you can set the following properties:

- **Capture DDLs:** When this option is selected, the following actions occur:
 - Operational artifacts are created (by Replicate) in the database when the task starts. In order to capture DDL events, Qlik Replicate creates various artifacts in the AWS Aurora Cloud for PostgreSQL database when the task starts. You can later remove these artifacts as described in *Removing Replicate artifacts from the AWS Aurora Cloud for PostgreSQL database (page 145)*.
 - Streamed DDL events are captured.
- **Create DDL artifacts in schema:** The schema in which the operational DDL database artifacts will be created. The default value is "Public".
- **WAL heartbeat** - An Apply Changes task that is running but not capturing changes (due to source table inactivity) will continue to occupy the LSN position in its replication slot, thereby preventing truncation of the WAL. Since the WAL is a server-wide resource used by all AWS Aurora Cloud for PostgreSQL processes, it may grow extremely large if no changes are captured for an extended period. To prevent this from happening, enable the "WAL heartbeat" option. When this option is enabled, the AWS Aurora Cloud for PostgreSQL source endpoint mimics task activity by periodically committing pseudo transactions (i.e. "Heartbeats") to the heartbeat table, thereby advancing the task slot's LSN position.
 - **Schema for heartbeat artifacts:** The schema in which the WAL heartbeat table (`attrep_wal_heartbeat`) will be created. The default value is "public".
 - **Heartbeat frequency (minutes):** The frequency with which to commit transactions to the heartbeat table.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

Removing Replicate artifacts from the AWS Aurora Cloud for PostgreSQL database

In order to capture DDLs, Qlik Replicate creates various artifacts in the AWS Aurora Cloud for PostgreSQL database when the task starts. When the task completes, you may wish to remove these artifacts.

To remove the artifacts, issue the following statements (in the order they appear below), where `public` is the default schema in which the artifacts were created:

- `drop event trigger attrep_intercept_ddl`



The event trigger does not belong to a specific schema.

- `drop function public.attrep_intercept_ddl()`
- `drop table public.attrep_ddl_audit`
- `drop schema public`
- `drop table schema.attrep_wal_heartbeat`

where `schema` is the **Schema for heartbeat artifacts** specified in the **Advanced** tab of the endpoint settings.



Dropping a schema should be done with extreme caution, if at all. Never drop an operational schema, especially not `public`.

9.2 Using Amazon RDS for SQL server as a source

This section describes how to set up and use an Amazon RDS for SQL Server database as the source database in a replication task.

In this section:

- *Prerequisites (page 147)*
- *Limitations and considerations (page 148)*
- *Required permissions (page 149)*
- *Supported compression methods (page 149)*
- *Amazon RDS for SQL server source data types (page 150)*
- *Non-supported data types (page 152)*
- *Homogeneous Replication (page 153)*
- *Preparing the Amazon RDS for SQL server database for Replication (page 154)*
- *Setting general connection properties (page 156)*
- *Setting advanced connection properties (page 158)*
- *Artifacts created by Replicate in the database (page 158)*

Prerequisites

Make sure that the following prerequisites have been met:

Qlik Replicate for Windows

Install Microsoft ODBC Driver 18.1 for Windows on the Qlik Replicate Server machine.

Qlik Replicate for Linux

The procedure below assumes that you have installed a single default instance of Replicate on Linux (areplicate). If you have [installed multiple instances](#), replace areplicate with the name of the instance running the task with a Microsoft SQL Server source. If several instances are running such as task, the procedure needs to be repeated for each instance.

Install Microsoft ODBC Driver 18.1 for Linux on the Replicate machine.

Then, open a Unix shell and do the following:

1. Change the working directory to `<Replicate-Install-Dir>/bin`.
2. Stop the Replicate service:

```
./areplicate stop
```
3. Optionally, confirm that the service has stopped:

```
./areplicate status
```
4. Copy the driver location to the `site_arep_login.sh` file as follows:

```
echo "export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/microsoft/msodbcsql18/lib64/" >> site_arep_login.sh
```

This will add the driver to "LD_LIBRARY_PATH" and update the driver location in the **site_arep_login.sh** file.
5. Optionally, confirm that the driver location was copied:

```
cat site_arep_login.sh
```
6. Start the Replicate service:

```
./areplicate start
```
7. Optionally confirm that the service has started:

```
./areplicate status
```
8. Replicate requires the following ODBC library: `libmsodbcsql-18.1.so.1.1`
To check which library version is currently installed Issue the following command:

```
ls /opt/microsoft/msodbcsql/lib64/
```

If the existing library has a different version number (e.g. `libmsodbcsql-18.0.so.1.1`), you need to create a symbolic link between the existing library and the required library.
To do this, issue the following commands:

```
cd /opt/microsoft/msodbcsql18/lib64/
ln -s existing_library_name libmsodbcsql-18.1.so.1.1
```

where *existing_library_name* is the name of the currently installed library (e.g. `libmsodbcsql-18.0.so.1.1`).

Additional prerequisites

- An Amazon RDS for SQL Server account with the specific access privileges is required. See [Source Permissions](#) for more information.

- Amazon RDS for SQL Server as a source must be configured for a full backup to work with Qlik Replicate. For more information, see *Preparing Amazon RDS for SQL server for backup and recovery* (page 155).

Limitations and considerations

When using an Amazon RDS for SQL Server source endpoint in a Replicate task, the following imitations apply:

- Amazon RDS for SQL Server moves the online logs to the backup location when the maximum allocated storage size is reached. This may sometimes occur before Replicate has read the logs, resulting in missing changes on the target. Therefore, in a system with a high volume of changes (or if you notice that some changes are not being captured), best practice is to increase the allocated storage size as described in [How can I troubleshoot storage consumption in my Amazon RDS DB instance that is running SQL Server?](#)
- A Secondary SQL Server database is not supported as a source database.
- The Amazon RDS for SQL Server Replication Publisher definitions for the database that was used in the task are not removed when you remove a task. an Amazon RDS for SQL Server system administrator must delete these definitions from Amazon RDS for SQL Server.
- Sparse tables are not supported.
- Replicating data from indexed views is not supported.
- Renaming tables using `sp_rename` is not supported (e.g. `sp_rename 'Sales.SalesRegion', 'SalesReg';`)
- Renaming columns using `sp_rename` is not supported (e.g. `sp_rename 'Sales.Sales.Region', 'RegID', 'COLUMN';`)
- TRUNCATE events will not be captured.
- Replication of computed columns is supported in Full Load replication tasks only. During CDC, INSERTs to computed columns will appear as NULL on the target and UPDATES will be ignored.
- Partition switching is not supported.
- When using the WRITETEXT and UPDATETEXT utilities, Qlik Replicate does not capture events applied on the source database.
- The following DML pattern is not supported:
`select <*> into <new_table> from <existing_table>`
- Column-level encryption is not supported.
- Amazon RDS for SQL Server backup to multiple disks is not supported.
- When inserting a value into SQL Server spatial data types (GEOGRAPHY and GEOMETRY), one can either ignore the SRID (Spatial Reference System Identifier) property - in which case the default SRID will be used (0 for GEOMETRY and 4326 for GEOGRAPHY) - or specify a different number. When replicating tables with spatial data types, Qlik Replicate replaces the SRID that was inserted by user with the default SRID.
- Replicating GEOGRAPHY and GEOMETRY data types is only supported in limited LOB mode.
- Columnstore indexes are not supported.
- Memory-optimized tables (using In-Memory OLTP) are not supported.
- Temporal databases are not supported
- Delayed durability is not supported

- Table change tracking is not supported
- Masked data will be created in the target without masking.
- Replication from databases configured to use ADR (Accelerated Database Recovery) is not supported.
- Capturing DDL and DML statements within a single transaction is not supported.
- Replicating a table or schema with an apostrophe (') in its name is not supported.
- UPDATES to a Primary Key/Unique Index that affect multiple rows may cause conflicts when applying the changes to the target. This may occur, for example, if the UPDATES are applied to the target as INSERT/DELETE operations rather than a single UPDATE operation. In such a scenario (where UPDATES to a Primary Key/Unique Index affect multiple rows), working in Batch optimized apply Change Processing mode is not supported as it may result in the table being ignored. Working in Transactional Apply Change Processing mode however, may result in constraint violations. If this happens, you either need to reload the relevant table or locate the problematic records in the *Apply exceptions* (page 1298) Control Table and edit them manually in the target database. See also: *Change Processing Tuning* (page 1181).
- Changes applied to the database using DACPAC will not be captured.
- System tables and views are not supported.

Non-supported Amazon RDS for SQL server security features

Tables that use the following Amazon RDS for SQL Server security features are not supported:

- Always Encrypted
- Row-Level Security

Required permissions

To use an Amazon RDS for SQL Server source in a Qlik Replicate task, the user specified in the Amazon RDS for SQL Server [endpoint connection settings](#) must be a member of the `db_owner` database role.

Supported compression methods

The table below lists which compression methods Qlik Replicate supports.

Supported compression methods

Compression method	Supported?
Row/Page Compression (at Partition Level)	Yes
Vardecimal Storage Format	No
Vardecimal Storage Format	No
Sparse Columns	No
Vardecimal Storage Format	No
Sparse Columns	No
Columnar Structure Compression	No

Amazon RDS for SQL server source data types

The Amazon RDS for SQL Server source for Qlik Replicate supports most Amazon RDS for SQL Server data types. The following table shows the Amazon RDS for SQL Server source data types that are supported when using Qlik Replicate and the default mapping to Qlik Replicate data types. Note that Amazon RDS for SQL Server data types are only mapped to Qlik Replicate data types when the target endpoint is *not* Amazon RDS for SQL Server. For information on data type mapping and collation support when the target endpoint is Amazon RDS for SQL Server, see *Homogeneous Replication (page 153)* below.

For information on how to view the data type that is mapped in the target, see the section for the target endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.



Collatable data types are indicated by an asterisk (*).

SQL server and Qlik Replicate data types

Amazon RDS for SQL server data types	Qlik Replicatedata types
BIGINT	INT8
BIT	BOOLEAN
DECIMAL	NUMERIC
INT	INT4
MONEY	NUMERIC (19,4)
NUMERIC (p,s)	NUMERIC
SMALLINT	INT2
SMALLMONEY	NUMERIC (10,4)
TINYINT	UINT1
REAL	REAL4
FLOAT	REAL8
DOUBLE	REAL8
DATETIME	DATETIME
DATETIME2	DATETIME
SMALLDATETIME	DATETIME
DATE	DATE
TIME	STRING (16)
DATETIMEOFFSET	STRING
*CHAR	STRING

9 Adding and managing source endpoints

Amazon RDS for SQL server data types	Qlik Replicatedata types
*VARCHAR	STRING
*VARCHAR (max) *TEXT To use this data type with Qlik Replicate, you must enable the use of CLOBs for a specific task. LOB columns for Amazon RDS for SQL Server tables are updated in the target even for UPDATE statements that did not change the value of the LOB column in Amazon RDS for SQL Server. During CDC, CLOB data types are supported only in tables that include a primary key. For more information, see LOB support in Task Settings/ <i>Metadata</i> (page 1166).	CLOB
*NCHAR	WSTRING
*NVARCHAR (length)	WSTRING
*NVARCHAR (max) *NTEXT To use this data type with Qlik Replicate, you must enable the use of NCLOBs for a specific task. LOB columns for Amazon RDS for SQL Server tables are updated in the target even for UPDATE statements that did not change the value of the LOB column in Amazon RDS for SQL Server. During CDC, NCLOB data types are supported only in tables that include a primary key. For more information, see LOB support in Task Settings/ <i>Metadata</i> (page 1166).	NCLOB
BINARY	BYTES
VARBINARY	BYTES

Amazon RDS for SQL server data types	Qlik Replicatedata types
<p>VARBINARY (max)</p> <p>IMAGE</p> <p>LOB columns for Amazon RDS for SQL Server tables are updated in the target even for UPDATE statements that did not change the value of the LOB column in Amazon RDS for SQL Server.</p> <p>To use this data type with Qlik Replicate, you must enable the use of BLOBs for a specific task.</p> <p>BLOB data types are supported only in tables that include a primary key.</p> <p>For more information, see LOB support in Task Settings/<i>Metadata</i> (page 1166).</p>	BLOB
TIMESTAMP	BYTES
UNIQUEIDENTIFIER	STRING
HIERARCHYID	STRING (250)
<p>XML</p> <p>LOB columns for Amazon RDS for SQL Server tables are updated in the target even for UPDATE statements that did not change the value of the LOB column in Amazon RDS for SQL Server.</p> <p>To use this data type with Qlik Replicate, you must enable the use of NCLOBs for a specific task.</p> <p>During CDC, NCLOB data types are supported only in tables that include a primary key.</p> <p>For more information, see LOB support in Task Settings/<i>Metadata</i> (page 1166).</p>	CLOB
GEOMETRY	CLOB
GEOGRAPHY	CLOB

Non-supported data types

Tables that include fields with the following data types are not supported by Qlik Replicate.

- CURSOR
- SQL_VARIANT
- TABLE



User-defined data types are supported according to their base-type. For example a user-defined data type based on DATETIME is handled as a DATETIME data type.

Homogeneous Replication

When replicating from an Amazon RDS for SQL Server source to an Amazon RDS for SQL Server target, most of the source and target data types will be identical. The exceptions are listed in the table below.



In homogeneous replication, the source data first passes through the Qlik Replicate data type and is therefore subject to any limitations of that type.

For information on Replicate data types and their limitations (where relevant), see *Replicate data types* (page 81).

For information on which Replicate data types the source data passes through when replicating from Amazon RDS for SQL Server, see the *Amazon RDS for SQL Server to Qlik Replicate data types mapping table* described earlier.



To prevent data truncation when replicating **XML** data types, it is strongly recommended to enable the [Allow unlimited LOB size](#) option in the task settings.

Additionally, in homogeneous replication, source column and table collations will be replicated to the target as described in *Column and table collation* (page 154).

Data type exceptions

When replicating from Amazon RDS for SQL Server database to a supported Microsoft SQL Server-based target, source and target data types are identical, with the following exceptions:

Data type comparisons between Amazon and Microsoft

Amazon RDS for SQL Server Source	Microsoft SQL Server Target
VARCHAR	VARCHAR (x) (when x=0 or x>8000) else VARCHAR (max)
NVARCHAR (length)	NVARCHAR (x) (when x=0 or x>8000) else NVARCHAR (max)

Amazon RDS for SQL Server Source	Microsoft SQL Server Target
VARBINARY	VARBINARY (x) (when x=0 or x>8000) else VARBINARY (max)
HIERARCHYID	STRING(250)
TIMESTAMP	VARBINARY

Column and table collation

When replicating from Amazon RDS for SQL Server to a supported Microsoft SQL Server-based target, column and table collations will be replicated to the target.



To support collation replication, the DBA must ensure that the collations defined for the source Amazon RDS for SQL Server database are the same as those defined for the supported Microsoft SQL Server-based target.

Non-nullable columns and primary/unique index names

Primary/Unique Index names are preserved during homogeneous replication. Non-nullable columns are also preserved during homogeneous replication, with the exception of the following data types:

- text
- ntext1
- varchar(max)
- nvarchar(max)
- varbinary(max)
- image
- xml

Preparing the Amazon RDS for SQL server database for Replication

This section describes the configuration requirements for using an Amazon RDS for SQL Server database. An Amazon RDS for SQL Server system administrator should carry out these tasks.

- *Preparing Amazon RDS for SQL server for backup and recovery (page 155)*
- *Defining Amazon RDS for SQL server database settings (page 156)*

Setting up Amazon RDS for SQL server for CDC (Change data capture)

The following section explains how to set up Amazon RDS for SQL Server for CDC.



When creating the Amazon RDS for SQL Server instance, full backup must be enabled, otherwise Replicate will not be able to capture changes.

To set up Amazon RDS for SQL Server for CDC:

1. Log in as the master user in the DB instance and set up the database for MS-CDC as follows:
`exec msdb.dbo.rds_cdc_enable_db 'db_name'`
where `db_name` is the name of the source database.
2. To enable MS-CDC, run the following for each of the intended source tables:



If you are running Full Load tasks, this step should only be run after Full Load completes.

```
exec sys.sp_cdc_enable_table
@source_schema = N'schema_name',
@source_name = N'table_name',
@role_name = NULL,
@supports_net_changes = 1
GO
```

where `schema_name` and `table_name` are the names of the source schema and the source table respectively.

3. Set the retention period for the changes to be available on the source by changing the `@pollinginterval` of the 'capture' job to the maximum possible value.

Limitations:

- CDC is supported using the transaction log only (i.e. backup is not supported).
- Events may be lost if they are moved to backup or truncated.

Preparing Amazon RDS for SQL server for backup and recovery

Qlik Replicate consumes changes captured from the database transaction log (TLOG). The TLOG is maintained by Amazon RDS for SQL Server for recovery purposes. All changes made to a database are written to the TLOG. The following happens when recovery is required:

- A backup copy of the database is made.
- Logged events are taken and used in a rolling-forward process where the recorded changes are replayed against that copy.

To prepare for backup and recovery you must make sure that the Amazon RDS for SQL Server Recovery Model is set up. This should be carried out by an Amazon RDS for SQL Server system administrator using the RDS management web interface.

The TLOG data is truncated as soon as it is no longer needed therefore the TLOG is not persistent. However, Qlik Replicate guaranteed delivery requires persistency in the changed data. To ensure persistency:

- A full database backup must be carried out before beginning to migrate data.
- The Recovery Model must be set to **Bulk logged** or **Full**.



After setting the Recovery Model, it is strongly recommended not to change it; doing so may result in loss of data.

Defining Amazon RDS for SQL server database settings

Set the following for the Amazon RDS for SQL Server database(s) that you are using as a source:

- Set the **Recovery model** to **Bulk logged** or **Full**. In this mode, the transaction Log is more durable and truncation occurs less frequently.
- Ensure that there is a full database backup for each Amazon RDS for SQL Server database that you are using as a source.
- When creating a connection string, it is possible to use any parameter supported by Amazon RDS for SQL Server. The Amazon RDS for SQL Server system administrator must ensure that the Amazon RDS for SQL Server instance is configured correctly so that the proper authentication credentials are accepted.

Working with Windows authentication

You can configure the Qlik Replicate Amazon RDS for SQL Server endpoint to log in to Amazon RDS for SQL Server (on Windows) using Windows authentication.

If you choose this option, you also need to make sure that:


- The Amazon RDS for SQL Server instance is set up to allow Windows log on.
 - The Qlik Replicate user is specified as the "Log on as" user for the "Qlik Replicate Server" service account.
- OR-
- Amazon RDS for SQL Server is configured to allow login for the Qlik Replicate Server service account.

Setting general connection properties


This section describes how to configure connection properties.

To add an Amazon RDS for SQL Server source endpoint to Qlik Replicate:

1. In the Qlik Replicate Console, click Manage Endpoint Connections to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints* (page 114).
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Amazon RDS for SQL Server database. This is optional.
4. Select **Source** as the database **role**.
5. Select **Amazon RDS for SQL Server** as the database **Type**.


 When this endpoint is used as a duplicated source in a Log Stream Staging setup, select the **Read changes from log stream** check box and then select the relevant Log Stream Staging task from the drop-down list.
For information on setting up and managing Log Stream Staging tasks, see *Using the Log Stream* (page 1072).

6. Specify the **Server name**. This is the host name or IP address of the computer with the Amazon RDS for SQL Server instance containing the source database.

 To override the default port, add the port to the server name, separated by a comma. For example, if the server name is `myserver.company.local` and the port is `3333`, then the server name should be entered like this:
`myserver.company.local,3333`

7. Select **Windows authentication** (only relevant when Replicate is installed on Windows) or **SQL Server authentication**.


If you select **Windows authentication**, the user credentials for the Windows domain will be used. This privilege must be configured in the Amazon RDS for SQL Server database by the system administrator. Note that this option is not relevant when Replicate Server is running on Linux.

 When using **Windows authentication**, make sure that the user account that is associated with the **Qlik Replicate Server** service has Network read and write permissions. This must be configured by a Windows system administrator.

See also *Working with Windows authentication* (page 156).

If you select **SQL Server authentication**, type the Amazon RDS for SQL Server authentication information (User name, Password) for the authorized user for this Amazon RDS for SQL Server database. If you do not know this information, see your Amazon RDS for SQL Server System Administrator.

To ensure that the password provided to Replicate is never used by any other principal, Replicate can be configured to automatically replace the user-entered password with a strong random password. For more information, see [Configuring Replicate to Automatically Replace the User-Entered Password](#).



- This information is case sensitive.
- To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**.
If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.
To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.



Make sure that the Amazon RDS for SQL Server user has the correct access privileges. For information on how to provide the required privileges, see [Required permissions \(page 149\)](#).

8. Type the **Database name** or click **Browse** and select one from the list of available databases. This is the name of the database from where you are replicating the data.

Setting advanced connection properties

In the **Advanced** tab, you can set internal parameters, set more options, and view a summary of your settings.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

Artifacts created by Replicate in the database

In order to capture changes, Qlik Replicate creates various artifacts in the **dbo** schema of the Amazon RDS for Microsoft SQL Server database when the task starts. You can safely remove them when the task completes.

The following artifact is created to track open transactions:

- #attrep_dbcc_open_trans_table

The following functions are created to verify that the LSN (Log Sequence Number) is valid:

- attrep_fn_LsnSegmentToHexa
- attrep_fn_NumericLSnToHexa

9.3 Using Amazon RDS for MySQL as a source

This section describes how to set up and use Amazon RDS for MySQL as a source in a replication task.

In this section:

- *Prerequisites (page 161)*
- *Limitations and considerations (page 162)*
- *Permissions (page 163)*
- *Data types (page 163)*
- *Setting general connection properties (page 167)*
- *Selecting a schema (page 169)*
- *Setting advanced connection properties (page 169)*

Prerequisites

Before using Amazon RDS for MySQL as an endpoint in a Replicate task, make sure the prerequisites outlined below have been met.

Driver prerequisites

Replicate on Windows

Install MySQL ODBC Unicode Driver 64-bit version 8.0.32 on the Replicate machine.

Replicate on Linux

1. Stop the Replicate service:
`./areplicate stop`
2. Optionally, confirm that the service has stopped:
`./areplicate status`
3. Install MySQL ODBC Unicode Driver 64-bit version 8.0.32 on the Replicate machine.
4. Change the working directory to `<Replicate-Install-Dir>/bin`.
5. Copy the driver location to the `site_arep_login.sh` file:
`echo "export LD_LIBRARY_PATH=\$LD_LIBRARY_PATH:/usr/lib64" >> site_arep_login.sh`
This will add the driver to "LD_LIBRARY_PATH" and update the driver location in the **site_arep_login.sh** file.
6. Optionally, confirm that the driver location was copied:
`cat site_arep_login.sh`
7. Make sure that the `/etc/odbcinst.ini` file contains an entry for MySQL, as in the following example:

```
[MySQL ODBC 8.0 Unicode Driver]
Driver = /usr/lib64/libmyodbc8w.so
UsageCount = 1
```
8. Start the Replicate service:
`./areplicate start`
9. Optionally confirm that the service has started:
`./areplicate status`

Amazon RDS for MySQL replication

Replication enables data from a primary Amazon RDS for MySQL database server to be copied to one or more secondary Amazon RDS for MySQL database servers.

Data can be ingested from either a primary database server or a secondary database server.

To replicate changes from a secondary database server (CDC), the binary logging parameter `log_slave_updates` needs to be set to `true` (1).

Setting up Amazon RDS for MySQL for CDC (Change Data Capture)

To set up Amazon RDS for MySQL or Amazon Aurora for MySQL for CDC:

1. Follow the instructions in the AWS Help for creating a new Parameter Group (see the Binary Logging Format section).
2. When creating the new Parameter Group, set: `binlog_format=row`
3. Save the new Parameter Group.

4. If you have an existing instance of Amazon RDS for MySQL, edit the instance to use the parameter specified in **Step 2** above. If you are provisioning a new instance of Amazon RDS for MySQL, reference the new Parameter Group created in **Step 1** above.
5. Finally, make sure that automated backup is enabled (the default).

Further reading: https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/USER_LogAccess.MySQL.BinaryFormat.html

Replicating 4-byte UTF8 emojis

Replication of 4-byte UTF8 emojis to certain targets requires the following preparation:

- **Microsoft SQL Server Target:** Transform the emojis from `WSTRING (n)` to `WSTRING (n*2)`.
- **Amazon Redshift Target:** Transform the emojis from `WSTRING (n)` to `WSTRING (n*2)`.

For information on defining transformations, see *Using the Transform tab (page 1086)*.

Limitations and considerations

The following limitations apply:

- The following DDLs are not supported:
 - All partition DDLs
 - Drop Table
 - Rename Table
- Changes that are older than one year will not be captured.
- Using the `alter table <table_name> add column <column_name>` statement to add columns to the beginning or to the middle of a table is not supported. When the `alter table <table_name> add column <column_name>` statement is used to add a column to the middle of a table, the column will be added to the end of the table instead.
- The **AR_H_USER** header column is currently not supported. For information on using header columns, see *Headers (page 1143)*.
- If an Amazon RDS for MySQL table contains LOBs and the task's **Replicate Lob columns** option is disabled, the table will be replicated without the LOB columns. Note that this only applies to MEDIUMBLOB, LONGBLOB, MEDIUMTEXT and LONGTEXT columns. This limitation does not apply to BLOB, TINYBLOB, TEXT and TINYTEXT columns.
- If the Amazon RDS for MySQL database is stopped during Full Load, the Full Load will end successfully, but the tables on the target may have less rows than the source tables. If this should happen, either restart the task or reload the tables with the missing rows.
- A Unique Index/Primary Key created on only part of the column data is not supported. If such a Unique Index/Primary Key is encountered, it will be created on the entire target column. The following is an example of a statement that creates a Unique Index/Primary Key using only part of the column data:

```
CREATE INDEX partial_name ON customer (name(10));
```
- Changes to computed (virtual) columns will not be captured.

- Replicate supports zero dates only if both the source and target MySQL-based databases are configured with the same "Strict Mode" (On or Off). If this is not the case, zero dates can be transformed to non-zero dates [using the transformation feature](#).
- The Transaction ID of XA transactions will not be displayed in the replication ChangeTable/Audit Table **transaction_id** header.
- Binary log transaction compression is not supported.
- The MySQL primary-primary configuration is not supported.
- Failover of cluster nodes is not supported.
- Capturing changes from an encrypted binlog is not supported.
- Due to an issue with MySQL ODBC Driver 8.0.027 or later, empty TEXT columns may not be replicated correctly during Full Load. For example, if one table row contains a TEXT column with a value and the same column in the next row contains an empty value (but not NULL), both rows will display the value of the first row on the target.

Permissions

The user specified in the endpoint connection settings must have the following permissions:

Full Load tasks

The ReplicationAdmin role with the SELECT privilege for the designated source tables.

Apply Changes tasks

The ReplicationAdmin role with the REPLICATION CLIENT and REPLICATION SLAVE privileges.

Checking for open XA transactions

When the task is initialized, Replicate checks whether there are any open XA transactions (transactions that use a two phase commit protocol). To allow this query to be performed, the following permission needs to be granted:

```
grant XA_RECOVER_ADMIN on *.* to 'userName'@'%';
```

Where `userName` is the user specified in the endpoint settings.

If this permission is not granted, the following error will be shown when you run the task:

```
Failed to check for open XA transactions. Verify that the user specified in the endpoint settings has the XA_RECOVER_ADMIN permission. Capturing changes while there are open XA transactions might lead to loss of data. If there are no XA transactions in this database, add the internal parameter IgnoreOpenXaTransactionsCheck to skip the XA transactions check.
```

Data types



The following table shows the Amazon RDS for MySQL database source data types that are supported when using Qlik Replicate and the default mapping to Qlik Replicate data types. When replicating to a MySQL target, the source and target data types are the same, apart from the exceptions described in *Homogeneous Replication* (page 166).

For information on how to view the data type that is mapped in the target, see the section for the target database you are using.


9 Adding and managing source endpoints

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

Amazon RDS for MySQL and Qlik Replicate data type comparison

Amazon RDS for MySQL source data types	Qlik Replicate data types
INT	INT4
BIGINT	INT8
MEDIUMINT	INT4
TINYINT	INT1
SMALLINT	INT2
UNSIGNED TINYINT	UINT1
UNSIGNED SMALLINT	UINT2
UNSIGNED MEDIUMINT	UINT4
UNSIGNED INT	UINT4
UNSIGNED BIGINT	UINT8
DECIMAL (10)	NUMERIC (10,0)
BINARY	BYTES (1)
BIT	BOOLEAN
BIT (64)	BYTES (8)
BLOB	BYTES (65535)
LOBLOB	BLOB
MEDIUMBLOB	BLOB
TINYBLOB	BYTES (255)
DATE	DATE
DATETIME	DATETIME
<div data-bbox="204 1473 874 1684" style="border: 1px solid gray; padding: 5px; margin-bottom: 5px;">  <i>DATETIME without a parenthetical value is replicated without milliseconds, whereas DATETIME with a value of 1-5 - e.g. DATETIME(5) - is replicated with milliseconds.</i> </div> <div data-bbox="204 1713 874 1886" style="border: 1px solid gray; padding: 5px;">  <i>When replicating a DATETIME column, the time remains the same on the target (i.e. it is not converted to UTC).</i> </div>	
TIME	STRING

9 Adding and managing source endpoints

Amazon RDS for MySQL source data types	Qlik Replicate data types
TIMESTAMP <div style="border: 1px solid black; padding: 5px; display: inline-block;">  <p>When replicating a <i>TIMESTAMP</i> column, the time is converted to UTC on the target.</p> </div>	DATETIME
YEAR	INT2
DOUBLE	REAL8
FLOAT If the FLOAT values are not in the range specified below, use a transformation to map FLOAT to STRING. For an explanation of how to do this, see <i>Using the Transform tab (page 1086)</i> . Supported FLOAT range: - 1.79E+308 to -2.23E-308, 0 and 2.23E-308 to 1.79E+308	REAL (DOUBLE)
*VARCHAR (45)	WSTRING (45)
*VARCHAR (2000)	WSTRING (2000)
*VARCHAR (4000)	WSTRING (4000)
VARBINARY (4000)	BYTES (4000)
VARBINARY (2000)	BYTES (2000)
*CHAR	WSTRING
*TEXT	WSTRING (65535)
*LONGTEXT	NCLOB
*MEDIUMTEXT	NCLOB
*TINYTEXT	WSTRING (255)
GEOMETRY	BLOB
POINT	BLOB
LINESTRING	BLOB
POLYGON	BLOB
MULTIPOINT	BLOB
MULTILINESTRING	BLOB

Amazon RDS for MySQL source data types	Qlik Replicate data types
MULTIPOLYGON	BLOB
GEOMETRYCOLLECTION	BLOB
ENUM	WSTRING (Length) Where "Length" is the longest value in the ENUM.
SET	WSTRING (Length) Where "Length" is the total of all values in the SET, including commas.
JSON	CLOB



If the DATETIME and TIMESTAMP data types are specified with a “zero” value (i.e. 0000-00-00), you need to make sure that the target database in the replication task supports "zero" values for the DATETIME and TIMESTAMP data types. If they are not supported, you can use a transformation to specify a supported value (e.g. 1970.) Otherwise, they will be recorded as null on the target.

Homogeneous Replication

The following section describes how Replicate handles replication between an Amazon RDS for MySQL source and the following targets: MySQL, Microsoft Azure Database for MySQL, and Google Cloud SQL for MySQL.



- *In homogeneous replication, the source data first passes through the Qlik Replicate data type and is therefore subject to any limitations of that type.
For information on Replicate data types and their limitations (where relevant), see Replicate data types (page 81).
For information on which Replicate data types the source data passes through when replicating from Amazon RDS for MySQL, see the Amazon RDS for MySQL to Qlik Replicate data types mapping table described earlier.*
- *When replicating a TIMESTAMP column, the time is converted to UTC on the target.*

Limitations and considerations

Homogeneous replication is not supported with the following Replicate configurations or operations:

- Apply Changes only tasks
- Stopping and resuming a task
- Starting a task from a timestamp
- When the **If target table already exists** in the task settings' **Full Load Settings** tab is set to **Do nothing**

Data types

When replicating to an Amazon RDS for MySQL target endpoint, the data types will be identical with the following exceptions:

Amazon RDS for MySQL source and target data types

Amazon RDS for MySQL source data types	Amazon RDS for MySQL target data types
JSON	LONGTEXT



When using data types from other database engines, the data type on the target will be the corresponding MySQL data type. For more information on using non-native data types, refer to the MySQL online Help.

Collation

When replicating from one Amazon RDS for MySQL endpoint to another, table and column collations will be replicated to the target. Collatable data types are indicated by an asterisk (*) in Table 11–3 above.

To support collation replication, the DBA must ensure that the collations defined for the source Amazon RDS for MySQL database are the same as those defined for the target Amazon RDS for MySQL database.

Non-nullable columns and primary/unique index names

Non-nullable columns and Primary/Unique Index names are preserved during homogeneous replication.

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.



You can also use Amazon RDS for MySQL files as a source. For more information, see Using the Qlik Replicate file channel (page 1061).

To add an Amazon RDS for MySQL source endpoint to Qlik Replicate:

1. In the Qlik Replicate Console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.
2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Amazon RDS for MySQL database. This is optional.
4. Select **Source** as the endpoint **role**.
5. From the **Type** drop-down list, select **Amazon RDS for MySQL**.



When this endpoint is used as a duplicated source in a Log Stream Staging setup, select the **Read changes from log stream** check box and then select the relevant Log Stream Staging task from the drop-down list.

For information on setting up and managing Log Stream Staging tasks, see *Using the Log Stream* (page 1072).

6. In the **Server** field, enter the host name or IP address of the computer on which the Amazon RDS for MySQL database is installed.
7. Optionally, change the default port (3306).
8. Type the Amazon RDS for MySQL authentication information (**Username, Password**) for the authorized user for this Amazon RDS for MySQL database. If you do not know this information, see your Amazon RDS for MySQL database Administrator (DBA).



- The **User Name, Password** are case sensitive.
- Passwords containing curly brackets are not supported i.e. {}.



Make sure that the Amazon RDS for MySQL user entered in the Amazon RDS for MySQL Authentication section has the correct access privileges. For information on how to provide the required privileges, see *Permissions* (page 163).

SSL authentication options

In the **Security** section, you can configure Replicate to connect to the Amazon RDS for MySQL database using SSL.

- **SSL Mode:** Select one of the following:
 - **Preferred** - Establishes an encrypted connection if the server supports encrypted connections, falling back to an unencrypted connection if an encrypted connection cannot be established.
 - **None** - Connect with a surname and password only.
 - **Required** - Establishes an encrypted connection if the server supports encrypted connections. The connection attempt fails if an encrypted connection cannot be established.
 - **Verify CA** - Similar to **Required**, but also verifies the server Certificate Authority (CA) certificate against the configured CA certificates. The connection attempt fails if no valid matching CA certificates are found.
 - **Verify Full** - Similar to **Verify CA**, but also performs host name identity verification by checking the host name the client (i.e. Replicate) uses for connecting to the server against the identity in the certificate that the server sends to the client. The client checks whether the host name that it uses for connecting matches the Common Name value in the server certificate. The connection fails if there is a mismatch.
- **Client certificate path** - The path to the client certificate requested by the server.

- **Private key path** - The path to the client private key file in PEM format.
- **CA path** - The path to the Certificate Authority (CA) that issued the client certificate file in PEM format.

Selecting a schema

You can choose which Amazon RDS for MySQL database to access. After configuring the Amazon RDS for MySQL source database connection settings, open the **Select Tables** dialog box (by clicking the **Table Selection** button on the right of the console) and select which schema to use from the **Schema** drop down list.

See also *Defining and managing tasks* (page 109).

Setting advanced connection properties

In the **Advanced** tab, you can set the following parameters:

- **Check binary log for new events every:** Specify how often to check the binary log for changes when the endpoints is idle.
- **Additional ODBC connection properties:** Specify any additional ODBC connection parameters that may be required.
- **Database timezone:** Select the timezone where the source database is located.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

9.4 Using Amazon RDS for PostgreSQL as a source

This section describes how to set up and use an Amazon RDS for PostgreSQL database as a source in a replication task.

In this section:

- *Prerequisites (page 171)*
- *Required permissions (page 172)*
- *Setting up Amazon RDS for PostgreSQL for CDC (Change data capture) (page 174)*
- *Limitations and considerations (page 174)*
- *Supported data types (page 175)*
- *Homogeneous Replication (page 179)*
- *Setting general connection properties (page 181)*
- *Setting advanced connection properties (page 182)*
- *Removing Replicate artifacts from the Amazon RDS for PostgreSQL database (page 183)*

Prerequisites

The following section lists the prerequisites for working with Qlik Replicate and an Amazon RDS for PostgreSQL source.

Client prerequisites

Replicate for Windows

PostgreSQL ODBC Driver `psqlodbc_13_xx_xxx-x64` must be installed on the Replicate machine.



Make sure that the `psql.exe` path (e.g. "`C:\Program Files\PostgreSQL\13xx\bin`") is added to the system `PATH`.

Replicate for Linux

Download PostgreSQL 13 ODBC driver, and client programs and libraries.

- **Example PostgreSQL ODBC file name:** `postgresql13-odbc-13.02.0000-1PGDG.rhel8.x86_64.rpm`
- **PostgreSQL client programs and libraries file name:** `postgresql13-13.8-1PGDG.rhel<version>.x86_64.rpm`

Then, open a shell prompt and do the following:

1. Stop the Replicate service:
`./areplicate stop`
2. Optionally, confirm that the service has stopped:
`./areplicate status`
3. Install PostgreSQL 13 client programs and libraries.
4. Install the ODBC driver for PostgreSQL 13.
5. Change the working directory to `<Replicate-Install-Dir>/bin`.
6. Copy the driver location to the `site_arep_login.sh` file:
`echo "export LD_LIBRARY_PATH=\$LD_LIBRARY_PATH:/usr/lib:/usr/lib64:/usr/pgsql-13/lib" >> site_arep_login.sh`
This will add the driver to "LD_LIBRARY_PATH" and update the driver location in the **site_arep_login.sh** file.
7. Optionally, confirm that the driver location was copied:
`cat site_arep_login.sh`
8. Makes sure that the `/etc/odbcinst.ini` file contains an entry for PostgreSQL, as in the following example:

```
[PostgreSQL Unicode(x64)]
Description = PostgreSQL ODBC driver
Driver = /usr/pgsql-13/lib/psqlodbc.so
Setup = /usr/pgsql-13/lib/psqlodbcw.so
UsageCount = 1
```
9. Start the Replicate service:
`./areplicate start`
10. Optionally, confirm that the service has started:
`./areplicate status`

Apply changes prerequisites

When the **Apply Changes** task option is enabled, the user specified in the Amazon RDS for PostgreSQL source endpoint's **General** tab must be granted super-user permissions.

Required permissions

The user specified in the **General** tab when *Setting general connection properties (page 181)* must be granted the following permissions in the Amazon RDS for PostgreSQL database:

- **For Full Load replication:** Standard SELECT on the source database
- **For Apply Changes replication:** See *Setting up Amazon RDS for PostgreSQL for CDC (Change data capture) (page 174)*.

Using an account without the "superuser" role

If you are not using an account with the **rds_superuser** role, you need to create several objects to capture Data Definition Language (DDL) events. Create these objects in the selected account and then create a trigger in the master user account.

To do this:

1. Choose a schema where you want the objects to be created. The default schema is public. Ensure that the schema exists and is accessible by the non-privileged account.
2. Create the table **attrep_ddl_audit** by running the following command:

```
create table <objects_schema>.attrep_ddl_audit
(
  c_key      bigserial primary key,
  c_time     timestamp,      -- Informational
  c_user     varchar(64),    -- Informational: current_user
  c_txn      varchar(16),    -- Informational: current transaction
  c_tag      varchar(24),    -- Either 'CREATE TABLE' or 'ALTER TABLE' or
                          'DROP TABLE'
  c_oid      integer,        -- For future use - TG_OBJECTID
  c_name     varchar(64),    -- For future use - TG_OBJECTNAME
  c_schema   varchar(64),    -- For future use - TG_SCHEMANAME. For now,
                          holds the current_schema
  c_ddlqry   text           -- The DDL query associated with the current
                          DDL event
);
```

3. Create the function **attrep_intercept_ddl** by running the following command:

```
CREATE OR REPLACE FUNCTION <objects_schema>.attrep_intercept_ddl()
  RETURNS event_trigger
LANGUAGE plpgsql
AS $$
  declare _qry text;
```



```
BEGIN
  if (tg_tag='CREATE TABLE' or tg_tag='ALTER TABLE' or tg_tag='DROP
TABLE') then
    SELECT current_query() into _qry;
    insert into <objects_schema>.attrep_ddl_audit
    values
    (
      default,current_timestamp,current_user,cast(TXID_CURRENT() as
varchar(16)),tg_tag,0,'',current_schema,_qry
    );
    delete from <objects_schema>.attrep_ddl_audit;
  end if;
END;
$$;
```

4. If you are logged in with a non-privileged account, log out of the non-privileged account and log in with an account that has the **rds_superuser** role assigned to it.



If the **attrep_intercept_ddl** stored procedure is not being created in the default schema, you need to specify the schema name in the **Create DDL artifacts in schema** field in the endpoint settings' **Advanced** tab.

For more information on the replication configuration parameters, see the PostgreSQL Help.

5. Create the event trigger **attrep_intercept_ddl** by running the following command:

```
CREATE EVENT TRIGGER attrep_intercept_ddl ON ddl_command_end
EXECUTE PROCEDURE <objects_schema>.attrep_intercept_ddl();
```

6. Grant the following permissions to the non-privileged account:

- GRANT INSERT ON attrep_ddl_audit to <non-privileged-user>;
- GRANT DELETE ON attrep_ddl_audit to <non-privileged-user>;
- GRANT USAGE ON attrep_ddl_audit_c_key_seq TO <non-privileged-user>;
- GRANT rds_replication to <non-privileged-user>;

7. Grant SELECT permission to the user:

```
GRANT SELECT ON ALL TABLES IN SCHEMA <SCHEMA_NAME> TO <non-privileged-user>;
```

8. Grant usage on the schema to the user:

```
GRANT USAGE ON SCHEMA <SCHEMA_NAME> TO <non-privileged-user>;
```

Setting up Amazon RDS for PostgreSQL for CDC (Change data capture)

You can use the AWS master user account for the PostgreSQL database instance as the user for the Amazon RDS for PostgreSQL source endpoint. The master user account has the required roles that allow it to set up Change Data Capture (CDC).

If you prefer not to use the master user account, the account you wish to use must have the **rds_superuser** role and the **rds_replication** role. The **rds_replication role** grants permissions to manage logical slots and to stream data using logical slots.

To enable logical replication for an Amazon RDS PostgreSQL database instance:

1. Create a new parameter group with the following configuration:
 - a. Set the **rds.logical_replication** parameter in the database parameter group to **1**.
 - b. **max_wal_senders** – The number of tasks that can run in parallel.
 - c. **max_replication_slots** – Every connection (i.e. task) creates a new slot; this parameter configures the amount of simultaneous connections that can be established to the database.
2. Link the Amazon RDS instance to the parameter group.

Limitations and considerations

The following limitations apply when using Amazon RDS for PostgreSQL as a source:

- The database name cannot include a semi-colon (;).
- Task names cannot start with "rds" as RDS is a reserved string on AWS.
- Both the source table and the corresponding target table must have an identical Primary Key. In the event that one of the tables does not have a Primary Key, the result of DELETE and UPDATE record operations will be unpredictable.
- The “Start Process Changes from Timestamp” run option is not supported.
- Replication of the Before Image is not supported.
- Replication of multiple tables with the same name but a different case (e.g. `table1`, `TABLE1` and `Table1`) may cause unpredictable behavior and is therefore not supported.
- Change processing of [CREATE | ALTER | DROP] table DDLs are supported, unless they are held in an inner function/procedure body block or in other nested constructs.

For example, the following change will not be captured:

```
CREATE OR REPLACE FUNCTION attu.create_distributors1() RETURNS void
LANGUAGE plpgsql
AS $$
BEGIN
create table attu.distributors1 (did serial PRIMARY KEY, name varchar(40)
NOT NULL);
END;
$$;
```

- Change processing of TRUNCATE operations is not supported.
- Deferred Constraints are not supported.
- **Partitioned tables**

- CDC is not supported.
- When using the *Parallel Load* (page 1098) feature, table segmentation according to partitions or sub-partitions is not supported.
- When using [filters](#) that do not contain a Primary Key, DELETE operations will not be captured.
- A DDL operation that occurs during Full Load, but that was not preceded by a DML operation will not be captured.
- PostGIS datatypes are not supported.
- Unique Indexes with CASE expressions are not supported.
- The SQL MERGE command introduced in PostgreSQL 15.x has not been certified.

Supported data types



The following table shows the Amazon RDS for PostgreSQL target data types that are supported when using Qlik Replicate and the default mapping to the Qlik Replicate data types.

For information about data type mapping in homogeneous replication, see *Homogeneous Replication* (page 179).


For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).



PostgreSQL and Qlik Replicate data types

Amazon RDS for PostgreSQL data types	Qlik Replicate data types
INTEGER	INT4
SMALLINT	INT2
BIGINT	INT8
NUMERIC	NUMERIC(28,6)
NUMERIC(p,s)	If precision is ≤ 38 , then: NUMERIC(p,s) If precision is ≥ 39 , then: STRING
DECIMAL	NUMERIC(28,6)

Amazon RDS for PostgreSQL data types	Qlik Replicate data types
DECIMAL(p,s)	<p>If precision is ≤ 38, then:</p> <p>NUMERIC(p,s)</p> <p>If precision is ≥ 39, then:</p> <p>STRING</p>
REAL	REAL4
DOUBLE	REAL8
SMALLSERIAL	INT2
SERIAL	INT4
BIGSERIAL	INT8
MONEY	<p>NUMERIC(38,4)</p> <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;"> <p> <i>The MONEY data type is mapped to FLOAT in Microsoft SQL Server.</i></p> </div>
CHAR	WSTRING (1)
CHAR(n)	WSTRING (n)
VARCHAR(n) <div style="border: 1px solid gray; padding: 10px; margin-top: 10px;"> <p> <i>VARCHAR without a length (n) is not recognized as a valid data type by target endpoints. Consequently, if a source column data type is set to VARCHAR without an explicit length, Replicate will set a default length of 8000 bytes.</i></p> <p><i>You can change the default by setting the following internal parameter to the required length:</i></p> <p><code>unboundedVarcharMaxSize</code></p> <p>See also Internal Parameters.</p> </div>	WSTRING (n)
TEXT	NCLOB

9 Adding and managing source endpoints

Amazon RDS for PostgreSQL data types	Qlik Replicate data types
BYTEA	BLOB
TIMESTAMP	DATETIME
TIMESTAMP (z)	DATETIME
<div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;">  <p><i>Replicate only supports ISO formatted textual DATE formats (the default). If other formats are used, an error will be generated. You can change the date format in the Amazon RDS for PostgreSQL.conf file or using the <code>PGDATESTYLE</code> environment variable. You can also change the date format at database level.</i></p> </div>	
DATE	DATE
TIME	TIME
TIME (z)	TIME
INTERVAL	STRING (128) - 1 YEAR, 2 MONTHS, 3 DAYS, 4 HOURS, 5 MINUTES, 6 SECONDS
BOOLEAN	STRING (5)
ENUM	STRING (64)
CIDR	STRING (50)
INET	STRING (50)
MACADDR	STRING (18)
BIT (n)	STRING (n)
BIT VARYING (n)	STRING (n)
UUID	STRING
TSVECTOR	CLOB
TSQUERY	CLOB
XML	CLOB
POINT	STRING (255) "(x,y)"
LINE	STRING (255) "(x,y,z)"
LSEG	STRING (255) "((x1,y1), (x2,y2))"

Amazon RDS for PostgreSQL data types	Qlik Replicate data types
BOX	STRING (255) "((x1,y1), (x2,y2))"
PATH	CLOB "((x1,y1), (xn,yn))"
POLYGON	CLOB "((x1,y1), (xn,yn))"
CIRCLE	STRING (255) "(x,y),r"
JSON	NCLOB
<div style="display: flex; align-items: flex-start;"> <div style="margin-right: 10px;">  </div> <div> <p><i>Reading and writing JSON data is performed using wide string (where each character occupies 2 bytes). Therefore, when replicating JSON columns to a PostgreSQL target, set the limited lob size to twice the maximum JSON length. So, for example, if the JSON data is 4 KB, set the limited lob size value to 8 KB.</i></p> </div> </div>	
JSONB	NCLOB
<div style="display: flex; align-items: flex-start;"> <div style="margin-right: 10px;">  </div> <div> <p><i>Reading and writing JSONB data is performed using wide string (where each character occupies 2 bytes). Therefore, when replicating JSONB columns to a PostgreSQL target, set the limited lob size to twice the maximum JSONB length. So, for example, if the JSONB data is 4 KB, set the limited lob size value to 8 KB.</i></p> </div> </div>	
ARRAY	NCLOB
COMPOSITE	NCLOB
INT4RANGE	STRING (255)
INT8RANGE	STRING (255)
NUMRANGE	STRING (255)
TSRANGE	STRING (255)
INT4MULTIRANGE	STRING (255)
INT8MULTIRANGE	STRING (255)
NUMMULTIRANGE	STRING (255)
TSMULTIRANGE	STRING (255)

Amazon RDS for PostgreSQL data types	Qlik Replicate data types
CHARACTER VARYING	If length is specified: WSTRING (LENGTH) If no length is specified: WSTRING (8000)
TINTERVAL	WSTRING(255)



The PostGIS data type is not supported.

Homogeneous Replication

The following section describes how Replicate handles replication from Amazon RDS for PostgreSQL to the following targets: PostgreSQL, Microsoft Azure Database for PostgreSQL, Google Cloud SQL for PostgreSQL, and Google Cloud AlloyDB for PostgreSQL.

In homogeneous replication, source column and table collations will be replicated to the target as described in *Column and table collation (page 180)*.



In homogeneous replication, the source data first passes through the Qlik Replicate data type and is therefore subject to any limitations of that type.

*For information on Replicate data types and their limitations (where relevant), see *Replicate data types (page 81)*.*

*For information on which Replicate data types the source data passes through when replicating from Amazon RDS for PostgreSQL, see the *Amazon RDS for PostgreSQL to Qlik Replicate data types mapping table* described earlier.*

Limitations and considerations


Homogeneous replication is not supported with the following Replicate configurations or operations:

- Apply Changes only tasks
- Stopping and resuming a task
- Starting a task from a timestamp
- When the **If target table already exists** in the task settings' **Full Load Settings** tab is set to **Do nothing**

Data type considerations and exceptions

When replicating from Amazon RDS for PostgreSQL database to a PostgreSQL-based target (e.g. Microsoft Azure Database for PostgreSQL), source and target data types are identical, with the following exceptions:

Data type exceptions in homogeneous replication

Amazon RDS for PostgreSQL source	PostgreSQL-based target
ENUM	STRING
COMPOSITE	STRING
NUMERIC	NUMERIC (28,6)
NUMERIC (p,s) : A numeric array with precision and scale.	NUMERIC: A numeric array without precision and scale.
TINTERVAL	<div style="border: 1px solid gray; padding: 5px;">  <p><i>As PostgreSQL have discontinued support for the TINTERVAL data type, tables with this data type will not be created on the target if the target database version does not support it.</i></p> </div>
INT4MULTIRANGE	STRING (255)
INT8MULTIRANGE	STRING (255)
NUMMULTIRANGE	STRING (255)
TSMULTIRANGE	STRING (255)

Column and table collation

When replicating from Amazon RDS for PostgreSQL to a PostgreSQL-based target, column and table collations will be replicated to the target.



To support collation replication, the DBA must ensure that the collations defined for the source Amazon RDS for PostgreSQL database are the same as those defined for the PostgreSQL-based target database.

Non-nullable columns and primary/unique index names

Non-nullable columns and Primary/Unique Index names are preserved during homogeneous replication.

Replication of Partitioned Tables

In order to replicate partitioned tables from Amazon RDS for PostgreSQL to a PostgreSQL-based target, you need to do the following:

1. Manually create the parent and child tables in the target database.
2. Define a separate task to replicate to those tables.

3. Set the **Truncate before loading** option in the *Full Load Settings* (page 1171).

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add an Amazon RDS for PostgreSQL endpoint source database to Qlik Replicate:

1. In the Qlik Replicate console, click **Manage Endpoint Connections** to open the **Manage Endpoint Connections** dialog box. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints* (page 114).
2. In the **Name** field, type a name for your Amazon RDS for PostgreSQL database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Amazon RDS for PostgreSQL database. This is optional.
4. Select **Source** as the database **role**.
5. Select **Amazon RDS for PostgreSQL** as the database **Type**.



*When this endpoint is used as a duplicated source in a Log Stream Staging setup, select the **Read changes from log stream** check box and then select the relevant Log Stream Staging task from the drop-down list.
For information on setting up and managing Log Stream Staging tasks, see *Using the Log Stream* (page 1072).*

6. In the **Server** name field, specify the host name of the database instance, for example:
`myinstance.123456789012.us-east-1.rds.amazonaws.com`
7. Optionally, change the default **Port** (5432).
8. Enter the Amazon RDS for PostgreSQL database authentication information (**Username, Password**) of an authorized Amazon RDS for PostgreSQL user. If you do not know this information, see your Amazon RDS for PostgreSQL database system manager.



Consider the following:

- *This information is case sensitive.*
- *To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**.
If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.
To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.*



Make sure that the Amazon RDS for PostgreSQL database user entered in the Amazon RDS for PostgreSQL database Authentication section has the correct access privileges.

9. Type the **Database** name or select one from the list of available endpoints. This is the name of the Amazon RDS for PostgreSQL database from which you are replicating data.

SSL authentication options

In the **Security** section, you can configure Replicate to connect to the Amazon RDS for PostgreSQL database using SSL.

- **SSL Mode:** Select one of the following:
 - **Disable** - Connect with a surname and password only.
 - **Allow** - Establish an encrypted connection if requested by the server.
 - **Prefer** - Establishes an encrypted connection if the server supports encrypted connections, falling back to an unencrypted connection if an encrypted connection cannot be established.
 - **Require** - Establishes an encrypted connection if the server supports encrypted connections. The connection attempt fails if an encrypted connection cannot be established.
 - **Verify CA** - Similar to **Required**, but also verifies the server Certificate Authority (CA) certificate against the configured CA certificates. The connection attempt fails if no valid matching CA certificates are found.
 - **Verify Full** - Similar to **Verify CA**, but also performs host name identity verification by checking the host name the client (i.e. Replicate) uses for connecting to the server against the identity in the certificate that the server sends to the client. The client checks whether the host name that it uses for connecting matches the Common Name value in the server certificate. The connection fails if there is a mismatch.
- **Client certificate path** - The path to the client certificate requested by the server.
- **Secret key path** - The path to the client private key file in PEM format.
- **CA path** - The path to the Certificate Authority (CA) that issued the client certificate file in PEM format.
- **CRL path** - The path to the CRL certificate. This file contains certificates revoked by certificate authorities. If the server certificate appears in this list, the connection will fail.
- **SSL compression** - Select this option to compress the data before it is encrypted.

Setting advanced connection properties

In the **Advanced** tab, you can set the following properties:

- **Capture DDLs:** When this option is selected, the following actions occur:
 - Operational artifacts are created (by Replicate) in the database when the task starts. In order to capture DDL events, Qlik Replicate creates various artifacts in the Amazon RDS for PostgreSQL database when the task starts. You can later remove these artifacts as described in *Removing Replicate artifacts from the Amazon RDS for PostgreSQL database (page 183)*.
 - Streamed DDL events are captured.
- **Create DDL artifacts in schema:** The schema in which the operational DDL database artifacts will be created. The default value is "Public".
- **WAL heartbeat:** An Apply Changes task that is running but not capturing changes (due to source table inactivity) will continue to occupy the LSN position in its replication slot, thereby preventing truncation of the WAL. Since the WAL is a server-wide resource used by all Amazon RDS for PostgreSQL processes,

it may grow extremely large if no changes are captured for an extended period.

To prevent this from happening, enable the "WAL heartbeat" option. When this option is enabled, the Amazon RDS for PostgreSQL source endpoint mimics task activity by periodically committing pseudo transactions (i.e. "Heartbeats") to the heartbeat table, thereby advancing the task slot's LSN position.

- **Schema for heartbeat artifacts:** The schema in which the WAL heartbeat table (`attrep_wal_heartbeat`) will be created. The default value is "public".
- **Heartbeat frequency (minutes):** The frequency with which to commit transactions to the heartbeat table.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

Removing Replicate artifacts from the Amazon RDS for PostgreSQL database

In order to capture DDLs, Qlik Replicate creates various artifacts in the Amazon RDS for PostgreSQL database when the task starts. When the task completes, you may wish to remove these artifacts.

To remove the artifacts, issue the following statements (in the order they appear below), where `public` is the default schema in which the artifacts were created:

9 Adding and managing source endpoints

- `drop event trigger attrep_intercept_ddl`



The event trigger does not belong to a specific schema.

- `drop function public.attrep_intercept_ddl()`
- `drop table public.attrep_ddl_audit`
- `drop schema public`
- `drop table schema.attrep_wal_heartbeat`

where `schema` is the **Schema for heartbeat artifacts** specified in the **Advanced** tab of the endpoint settings.



Dropping a schema should be done with extreme caution, if at all. Never drop an operational schema, especially not `public`.

9.5 Using ARC CDC solutions in Qlik Replicate

This section describes how to use an ARC (Attunity Replicate Connect) CDC Solution as a Qlik Replicate endpoint.



- *For better performance, it is strongly recommended that Replicate runs on Windows when working with an ARC-based source.*
- *For all ARC sources, it is strongly recommended to install the ARC Agent in the same data center as the Replicate server.*

In this section:

- *Prerequisites for using ARC CDC solutions (page 186)*
- *ARC CDC solution security considerations (page 187)*
- *Limitations (page 189)*
- *ARC source data type mapping (page 189)*
- *Working with ARC CDC solutions (page 190)*
- *Setting advanced connection properties (page 193)*

Prerequisites for using ARC CDC solutions

You must have the following installed on your network and database to use ARC CDC solution.

- **ARC version 5.5 or later:** This must be installed on the same computer as the database you are using. You will need the installation kit for the computer platform that your database runs on. For example, if you are using an IBM IMS database, install ARC on the same mainframe computer where your IBM IMS database is located. For information on how to install ARC, see the *Attunity Replicate Connect Installation Guide* for the computer platform that is relevant to the CDC Solution you are working with.
- **Attunity Studio version 5.3.2 or later:** Attunity Studio is used to set up a CDC Solution. This will create the CDC Solution that can be used in the
- **When the ARC database is on DB400-AS4002:** To apply deletes to the target, journaling must be set to *BOTH.
- **ARC relational data sources that support table ownership:** If the table owner contains an underscore, you must create the ARC solution with a default table owner.
- **When the source endpoint is IBM IMS (ARC):** The ARC IMS Bulk data source is always created as IMS-DLI. You should specify the correct ARC IMS **Bulk started task** in the endpoint settings. The ARC USERLIB library contains the following started task examples:
 - NVIMSSRV for IMS DLI access
 - NVBMPSRV for IMS BMP access

Additional prerequisites when using ARC non-relational sources

The following prerequisites apply when using any of the *Using ARC CDC agents as endpoints (page 81)*.

If the source tables contain Primary Keys, you must edit the source table metadata in Attunity Studio. Mark the **Primary Key** columns as shown in the figure below by checking off all necessary checkboxes. This should be done at the start of creating the CDC solution when importing the tables/files.

Table Information

Name	Data Type	Size	Scale	Dimension	Offset	Fixed Offset	Primary Key
STUDENT_ID	int4			0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>
FIRST_NAME	string	32		0	4	<input type="checkbox"/>	<input type="checkbox"/>
LAST_NAME	string	32		0	36	<input type="checkbox"/>	<input type="checkbox"/>
DATE_OF_BIRTH	int_date			0	68	<input type="checkbox"/>	<input type="checkbox"/>
NUMOF_COURSES	int4			0	72	<input type="checkbox"/>	<input type="checkbox"/>
▲ COURSES	group			0	76	<input type="checkbox"/>	
▲ COURSE	group			8	0	<input type="checkbox"/>	
COURSE_ID	int4			0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>
COURSE_TITLE	string	48		0	4	<input type="checkbox"/>	<input type="checkbox"/>
INSTRUCTOR_ID	int4			0	52	<input type="checkbox"/>	<input type="checkbox"/>
NUMOF_ASSIGNMENTS	int4			0	56	<input type="checkbox"/>	<input type="checkbox"/>
▲ ASSIGNMENT	group			4	60	<input type="checkbox"/>	
ASSIGNMENT_TYPE	string	12		0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>
ASSIGNMENT_TITLE	string	48		0	12	<input type="checkbox"/>	<input type="checkbox"/>
DUE_DATE	int_date			0	60	<input type="checkbox"/>	<input type="checkbox"/>
GRADE	decimal	2	1	0	64	<input type="checkbox"/>	<input type="checkbox"/>
NUMOF_BOOKS	int4			0	2668	<input type="checkbox"/>	<input type="checkbox"/>
▲ BOOK	group			5	2672	<input type="checkbox"/>	
ISBN	string	10		0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>
RETURN_DATE	int_date			0	10	<input type="checkbox"/>	<input type="checkbox"/>

For more information on creating ARC Solutions, please refer to the *Attunity Replicate Connect User Guide and Reference*.



For information about installing the database you are working with, see the installation guide for that database.

ARC CDC solution security considerations

For an explanation of the security configurations and permissions necessary, see the CDC Solution reference in the *Attunity Replicate Connect User Guide and Reference*.

Encrypting communications between Replicate and ARC data sources

You can encrypt sessions between Replicate and ARC data sources. When a session is encrypted, all communications between Replicate and the selected ARC data source will be encrypted using AES-256 bit encryption. When capturing changes from a relational data source, the encryption key needs to be defined in

9 Adding and managing source endpoints

two locations: The Qlik Replicate ARC database and the ARC Agent machine. However, when capturing changes from a nonrelational database, the encryption key needs to be defined in four different locations: The Qlik Replicate ARC database, the ARC Agent machine, the ARC Router machine, and the Router Authenticator.

To encrypt communications between Replicate and ARC data sources:

1. On the **Agent** machine, create an encryption key as follows:
 1. Open Attunity Studio in **Design** view.
 2. In the **Configuration** tab, expand the machine on which your ARC Solution's Agent is installed.
 3. Expand the **Users** folder and select **NAV**.
The **User: NAV** tab opens.
 4. To the right of the **Encryption Keys** list (in the lower half of the screen), click the **Add** button.
The **Encryption Key** dialog opens.
 5. Enter an encryption key name and value and then click **OK**.



Steps 2-4 apply to non-relational ARC data sources only (e.g. VSAM). If you are working with a relational ARC data source, continue to Step 5.

2. On the **Router** machine, create an encryption key which has the same values as the encryption key that you created on the **Agent** machine. The procedure is the same as described in **Step 1**, but instead of expanding the machine on which your ARC Solution's Agent is installed, expand the machine on which your ARC Solution's Router is installed.
3. On the **Router** machine, define the Agent as an authenticator according to the following steps:
 1. In the **Configuration** tab, expand the machine on which the Router is installed. Then, right-click your solution's Router binding (e.g vsam_router) and select **Open**.
 2. In the **Machines** tab, click the **Security** button.
The **NAV** tab opens.
 3. To the right of the **Authenticators** list, click the **Add** button.
The **Add Authenticator** dialog box opens.
 4. From the **Resource type**, drop-down list, select **Adapter**.
 5. In the **Resource name** field, specify the name of your solution's Agent as it appears under the **Adapters** folder (e.g VSAM_ag).
 6. At the bottom of the dialog box, select the **Encryption key** check box and then specify the encryption key name and value in the designated fields. These values must be the same as the encryption key values defined in **Step 1**.
4. In the Router's **Properties** tab, expand the **comm** property and set the **defaultEncryptionMethod** property to **AES**.



*If the **Properties** tab is not displayed, open the **Preferences** dialog box (by selecting **Preferences** from the **Windows** menu), navigate to **Studio** and then select the **Show advanced environment parameters** option in the **Advanced** tab.*

5. In the **Advanced** tab of the Replicate ARC database, specify the encryption key name and value. These values must be the same as the encryption key values defined in **Step 1**.

9 Adding and managing source endpoints

For more information on the Advanced tab, see *Using ARC CDC agents as endpoints (page 81)*.
See also: *Using ARC CDC agents as endpoints (page 81)*.

Limitations

When working with ARC data sources, the following limitations apply:

- **IBM DB2 on iSeries (ARC):** Table and field names that contain the "/" character are not supported.
- Only one Replicate task can work with the same ARC Agent concurrently.
- Replication of DDL changes to the target endpoint is not supported.
- Replication of data with 4-byte emoji characters is not supported with IBM IMS.

ARC source data type mapping

The table below shows the ARC source data types that are supported when using Qlik Replicate and the default mapping to Qlik Replicate data types.

For an explanation of the supported data types for the ARC CDC Solution you are using, see the CDC Solution reference in the **Attunity Replicate Connect User Guide and Reference**.

For information on how to view the data type that is mapped in the target, see the section for the target database you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

AIS: ARC and Qlik Replicate data types

ARCData Types	Qlik ReplicateData Types
INT	REAL4
REAL	REAL4
FLOAT	REAL8
BIT	INT1
TINYINT	INT1
SMALLINT	INT2
BIGINT	NUMERIC
DECIMAL	NUMERIC
NUMERIC	NUMERIC
MONEY	NUMERIC
SMALLMONEY	NUMERIC
DATETIME	DATETIME
SMALLDATETIME	DATETIME
CHAR	STRING

ARCData Types	Qlik ReplicateData Types
VARCHAR	STRING
NCHAR	STRING
NVARCHAR	STRING
BINARY	BYTES
VARBINARY	BYTES
TIMESTAMP	BYTES
UNIQUEIDENTIFER	STRING

Working with ARC CDC solutions

To use a CDC Solution from the Attunity Integration Suite, you must first create a CDC Solution in Attunity Studio. Then create a new database using the CDC Solution you created as the Qlik Replicate database. You can then use this database as your source for any task that you create. To use ARC CDC Solutions, carry out the following:

- *Create an ARC CDC solution in attunity replicate connect studio (page 190)*
- *Add the ARC data source to Qlik Replicate (page 191)*
- *Add the ARC CDC solution endpoint to a task (page 193)*

Create an ARC CDC solution in attunity replicate connect studio

Before you can begin to work with an ARC CDC Solution in Qlik Replicate, you must create a CDC solution using one of the supported ARC CDC Solutions using Attunity Replicate Connect Studio. For information on the required ARC installation necessary to create a CDC solution, see *Prerequisites for using ARC CDC solutions (page 186)*.

To create a CDC solution in Attunity Replicate Connect Studio:

1. Using Attunity Replicate Connect Studio, create a CDC Solution using the CDC Solution that you want to use as your source database in Qlik Replicate.
For information on creating a CDC Solution, refer to the Attunity Integration Suite User Guide and Reference.
2. At the end of the process for creating a CDC solution, you must deploy the solution. Do not activate the solution. Qlik Replicate activates the solution automatically when you begin to work with the CDC Solution.



If you activate the solution, then disable the router and staging area workspaces and keep the agent workspace enabled. For more information, see the Attunity Replicate Connect User Guide and Reference.

Add the ARC data source to Qlik Replicate

The next step is to add the ARC Data Source to Qlik Replicate. You do this by adding a database and selecting one of the supported ARC database types.

If you selected one of the supported relational data sources, continue from *Adding a relational ARC data source to Qlik Replicate (page 191)*.

If you selected one of the supported non-relational data sources, continue from *Adding a non-relational ARC data source to Qlik Replicate (page 192)*.

See also *Additional prerequisites when using ARC non-relational sources (page 186)*.

For information on how to add endpoints, see *Defining and managing endpoints (page 114)*.

Adding a relational ARC data source to Qlik Replicate

To add a relational ARC data source to Qlik Replicate:

1. In the Qlik Replicate console, click **Manage Endpoint Connections** to open the **Manage Endpoint Connections** dialog box and then click **New Endpoint Connection**. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the ARC CDC Solution. This is optional.
4. Select **Source** as the **role**.
5. Select a relational ARC data source from the **Type** list. The ARC data sources are listed as **<Data Source> (ARC)**. For a list of supported relational data sources, see *Using ARC CDC agents as endpoints (page 81)*.



*When this endpoint is used as a duplicated source in a Log Stream Staging setup, select the **Read changes from log stream** check box and then select the relevant Log Stream Staging task from the drop-down list.
For information on setting up and managing Log Stream Staging tasks, see *Using the Log Stream (page 1072)*.*

6. In the **Host/IP** field, type the name or IP Address of the computer where the CDC Solution (data source) you defined in Attunity Studio is located.
7. In the **Port** field, type the port number for the port you used when creating the CDC Solution in Attunity Studio. The default port number is 2551.
8. In the **CDC Solution** field, enter the name of the solution you defined when you created the data source in Attunity Studio.
9. In the **User name** and **Password** fields, enter the username and password required to access the database.
10. Click **OK** to add the database to Qlik Replicate. You can use this database as the source database for any replication task that you create.



To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**.

If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

Adding a non-relational ARC data source to Qlik Replicate

When you add a database to Qlik Replicate and you select a non-relational ARC data source as the database type, the following dialog box opens.

To add an ARC source database to Qlik Replicate:

1. In the Qlik Replicate console, click **Manage Endpoint Connections** to open the **Add Endpoint Connections** dialog box and then click **New Endpoint Connection**. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the ARC CDC Solution. This is optional.
4. Select **Source** as the **role**.
5. Select an ARC non-relational data source from the **Type** list. The ARC data sources are listed as **<database> (ARC)**, for example **RMS (ARC)**. For a list of supported non-relational data sources, see *Using ARC CDC agents as endpoints (page 81)*.



When this endpoint is used as a duplicated source in a Log Stream Staging setup, select the **Read changes from log stream** check box and then select the relevant Log Stream Staging task from the drop-down list.

For information on setting up and managing Log Stream Staging tasks, see *Using the Log Stream (page 1072)*.

6. When working with the **RMS (ARC)** data source, choose one of the following Change Processing modes:
 - **Non relational** (the default) - When this mode is selected, Replicate reads the changes from a CSV file that contains the modified data records. Use this mode if you need to retrieve changes to arrays and variant tables.
If you select this option, continue from Step 7 below.
 - **Relational** - When this mode is selected, Replicate reads the changes directly from the ARC Agent. Relational mode improves performance but does not support changes to complex data structures such as arrays and variant tables.
If you select this option, continue from Step 6 in *Adding a relational ARC data source to Qlik Replicate (page 191)*.
7. In the **Port** field, type the port number for the port you used when creating the CDC Router in Attunity Studio. The default port number is 2551.

8. In the **CDC Solution** field, enter the name of the solution you defined when you created the data source in Attunity Studio.
9. If a username and password are required to access the CDC Solution Router, enter them in the **User name** and **Password** fields in the **Local ARC router** section.
10. If a username and password are required to access the CDC Solution, enter them in the **User name** and **Password** fields in the **ARC on <source> machine** section.
11. Required for **IBM IMS (ARC)** only: In the **Bulk started task** field, specify the correct z/OS Started Task name for IMS/BMP or IMS/DLI. This member was copied to the z/OS PROCLIB library from <ARC HLQ>.USERLIB. NVBMPSRV and NVIMSSRV are the provided member names.



If you choose IMS/DLI, you will need to close the database to IMS/TM or IMS/DBCTL. This option might be faster than using BMP. IMS/BMP does not require exclusive access to the database.

12. Click **OK** to add the database to Qlik Replicate. You can use this database as the source database for any replication task that you create.



*To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**. If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box. To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.*

Add the ARC CDC solution endpoint to a task

You can use any ARC CDC Solution that you define as the source in a task. To use an ARC CDC Solution as your source, drag the ARC database from the **Endpoints** pane to your task.

For information on how to create a task, see *Adding a source and target endpoint to a task (page 117)*.

Setting advanced connection properties

In the **Advanced** tab, you can set advanced properties.

- **Encryption key name:** Enter name of the encryption key defined in the **User: NAV** tab in ARC.
- **Encryption key value:** Enter value of the encryption key specified in the **Encryption key name** field above.



*For a detailed explanation of how to encrypt session between Replicate and ARC endpoints, see *Encrypting communications between Replicate and ARC data sources (page 187)*.*

- **Fixed NAT:** Select this to indicate that the connection is made with a fixed network address translation.

- **Timeout:** Enter the amount of time, in seconds, to wait for interactions before disconnecting. 0 indicates that the system does not timeout. The default value is **0**.
- **Event wait:** Enter the maximum amount of time (in seconds) to wait for a change event to take place before the system times out. The default value is **300**.
- **CDC batch size:** Enter the maximum number of change events that can be transferred in a single batch. The default value is **200**.
- **Bulk batch size:** Enter the unloading batch size. The default value is **100**.
- **Trace:** Select this to enable tracing for the change processing.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

9.6 Using a file as a source

This section describes how to set up and use delimited text files as a source in a replication task. You can use the File target endpoint to export database tables to files, which can then be used as a source in a Replicate task with a File source endpoint.

In this section:

- *General overview (page 196)*
- *File source overview (page 196)*
- *Prerequisites (page 199)*
- *Limitations (page 199)*
- *Setting general connection properties (page 199)*
- *Setting advanced options (page 207)*

General overview

The Replicate File endpoint can be used either as a source or as a target. When used as a source, the File endpoint requires the source files to be in delimited text file format. When used as a target, the File endpoint generates the data files either in delimited text file format (CSV) or in JSON format (according to the format selected in the endpoint settings).

Delimited text files are used to store data in tabular format. Examples of delimited text file formats include the CSV (Comma Separated Values) and TSV (Tab Separated Values) formats. Some organizations may implement procedures that export data from a database to a delimited text file while others may simply prefer this format as a convenient way of storing tabular data.

In a delimited text file, each record in the table occupies a separate row. Delimiters are used to mark the beginning of a new row or the beginning of a new column. Virtually any character can be used as a delimiter, although a newline (\n) is often used to separate rows, and commas are commonly used to separate columns.

In JSON files, each record is represented by a single line.

So, for example, the following table:

JSON data format example

book_id	title	price	is_hardcover
123	Angels	6.99	false
456	The Fallen	6.49	true
789	Rise Up	7.23	true

Will be represented as:

```
{ "book_id": 123, "title": "Angels", "price": 6.99, "is_hardcover": false }  
{ "book_id": 456, "title": "Fallen", "price": 6.49, "is_hardcover": true }  
{ "book_id": 789, "title": "Rise Up", "price": 7.23, "is_hardcover": true }
```

See also *File source overview (page 196)* and *File target overview (page 698)*.

File source overview

When you configure the File Source endpoint in Replicate, you need to specify which row and column delimiters are used in your CSV source files as well as which characters are used to enclose columns containing delimiters. You may also need to specify which character is used to escape columns enclosed in double-quotes (should such columns exist in your source files). If you have previously used Replicate to transfer endpoint tables to files and you now wish to use those files as a source in another task, then you should specify the same delimiters (i.e. that were used to generate target files).

For more information, see *Setting general connection properties (page 199)*.

Three types of delimited files are used in the Replicate Source File endpoint:

- *Full load files (page 197)*
- *Change files (page 198)*
- *Reference files (page 197)*

Reference files

Change Files can either reside in a single location or in multiple locations. To access Change Files that reside in different locations and/or that do not include the target table names, you need to use a Reference File. If the rows in the Change File(s) contain the names of the tables to which to apply the changes, then the Reference File only needs to contain the paths to the Change Files. If each Change File contains changes for a single table, then each of the Change File paths in the Reference File needs to be preceded by the name of its corresponding target table.

For more information on Reference File and Reference File formats, see [Change Processing](#).

Each row in the Reference File should be formatted as follows:

```
[<table_name>],<full path to Change File>
```

Where [`<table_name>`] is required only if the referenced Change File contains changes for a single table.



Reference File names cannot exceed 70 characters (no such limitation exists for the path length). Reference File names that exceed 70 characters will be ignored and appropriate warning will be written to the log.

Reference File: Each Change File contains changes for a single table:

```
table1,c:\temp\cdc1.csv
```

```
table2,c:\temp\cdc2.csv
```

```
table3,c:\temp\cdc3.csv
```

Reference File: Each Change File contains changes for multiple tables:

```
c:\temp\cdc1.csv
```

```
c:\temp\cdc2.csv
```

```
c:\temp\cdc3.csv
```

Full load files

Full Load Files are used to populate the empty source tables with data. Full Load Files should only contain the table data. The source tables themselves are created using the External Table Editor provided in the File source endpoint configuration. Both the tables and their data are replicated to the target endpoint during the Full Load stage of the Replicate task.

Example of a Full Load Data File

```
22,January,2014,male,5463565
```

```
12,May,2011,female,3236776
```

9, March, 2009, male, 9648675

For more information on Full Load data files and creating tables, see *Defining tables and full load data* (page 204).

Change files

A Change File is a delimited text file that contains a record of DML changes - represented as rows - to apply to the specified target tables. Replicate reads the Change File(s) and applies the changes to the relevant target tables, which can either be specified in the Change File itself or in a Reference File (see *Reference files* (page 197) below for details). Change Files are picked up from the source directory according to their modification date, thereby ensuring that the changes will be processed in the proper sequence.



The Change File modification date must be both newer than the task start timestamp and newer than the last processed Change File.

Each row in a Change File consists of the following delimited columns:

- (Optional) The change operation e.g. DELETE. If the operation field is absent, INSERT is assumed.
- The name of the target table to which to apply the change (only required if the Change File contains changes for multiple tables)
- (Optional) The timestamp of the change i.e. when the change occurred
- (Optional) The user who applied the change
- The data to change (one or more columns)

Change Files can either contain changes for multiple tables or for a single table, as shown in the examples below.



*To access Change Files that reside in different locations and/or that do not include the target table names, you need to use a Reference File. For more information on Reference Files, see *Reference files* (page 197).*



Change File names cannot exceed 70 characters (no such limitation exists for the path length). Change File names that exceed 70 characters will be ignored and appropriate warning will be written to the log.

Change File that contains changes for multiple tables

```
INSERT,table1,ts1,user,dog,cat,bird
```

```
INSERT,table2,ts1,user,dog,cat,bird
```

```
DELETE,table3,ts1,user,dog,cat,bird
```

Change File that contains changes for a single table

```
INSERT,,ts1,user,dog,cat,bird
```

INSERT,,ts1,user,dog,cat,bird

DELETE,,ts1,user,dog,cat,bird

Prerequisites

Before you begin to work with a File as a source in Qlik Replicate, make sure that the following prerequisites have been met:

- Qlik Replicate installed in your network
- Change Files, Full Load files and Reference Files should be in delimited text file format
- Source files (including the Reference File) should be accessible from the Qlik Replicate machine.

Limitations

The following limitations apply to the File source:

- Change Files that are currently being used in a Replicate task cannot be modified while the task is in progress.
- Stopping a Full Load task and then starting it again will start the task from the beginning (and not from the point at which it was stopped).
- Reload from timestamp is not supported
- Reloading a table during Change Processing is not supported

Setting general connection properties

You can add a File endpoint to Qlik Replicate to use as a source. For information on how to add endpoints, see *Defining and managing endpoints (page 114)*.

To add a File source endpoint to Qlik Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button.
2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the endpoint being used.
3. In the **Description** field, type a description that helps to identify the File endpoint. This is optional.
4. Select **Source** as the endpoint **role**.
5. Select **File** as the endpoint **Type**.



*When this endpoint is used as a duplicated source in a Log Stream Staging setup, select the **Read changes from log stream** check box and then select the relevant Log Stream Staging task from the drop-down list.*

*For information on setting up and managing Log Stream Staging tasks, see *Using the Log Stream (page 1072)*.*




6. Configure the settings in the **General** tab as described in the table below.

9 Adding and managing source endpoints


General tab settings

Option	Description
File Format	-
Field Delimiter	<p>The delimiter used to separate columns in the source files. The default is a comma.</p> <p>Example:</p> <p>mike,male</p>
Record delimiter	<p>The delimiter used to separate records (rows) in the source files. The default is a carriage return (\n).</p> <p>Example (Using an asterisk as the row delimiter)</p> <p>mike,male*sara,female</p>
Null value	<p>The character used to indicate a null value in the source files.</p> <p>Example (where * is the row delimiter and @ is the null value):</p> <p>mike,male,295678*sara,female,@</p>
Quote character	<p>The character used at the beginning and end of a column that contains the column delimiter character. The default is the double-quote character ("). When a column that contains column delimiters is enclosed in double-quotes, the column delimiter characters are interpreted as actual data, and not as column delimiters.</p> <p>Example (where a comma is the column delimiter):</p> <p>"sunroof, power-steering"</p>
Escape character	<p>The character used to escape a string when both the string and the column containing the string are enclosed in quotation marks. Note that the string's quotation marks will be removed unless they are escaped.</p> <p>Example (where " is the quote character and \ is the escape character):</p> <p>1955,"old, \"rare\", Chevrolet",\$1000</p>

9 Adding and managing source endpoints

Option	Description
Code page	<p>Specify the code page of your source files if it is different from the default (65001).</p> <div data-bbox="671 394 1386 600" style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p> <i>Windows and Linux systems use different code page conventions. The specified code page must comply with the code page convention of the source file system.</i></p> </div>
Ignore records	<p>Optionally, specify which header and footer rows in the source files to ignore. Make sure that the header and footer rows to ignore do not contain actual data.</p> <div data-bbox="671 759 1386 893" style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p> <i>The specified number will be ignored in Full Load tasks only.</i></p> </div>
<p>Change Processing</p> <div data-bbox="204 965 635 1328" style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p> <i>Changes cannot be captured from Change Files that are present during the Full Load operation. Consequently, the Change Files should be placed in their source location(s) only after Full Load completes.</i></p> </div>	-
Folder	<p>Select this option if your Change Files contain the target table names and reside in a single folder. Then specify the folder location in the designated field. You can optionally use wildcard characters to only process files that match the specified pattern.</p> <p>Example:</p> <p><code>c:\temp*changes.CSV</code></p> <p>See also: <i>Change files (page 198)</i>.</p>

9 Adding and managing source endpoints

Option	Description
Use Reference Files	<p>Select this option if you are using Reference Files to point to the location of the Change Files. Then specify one of the following:</p> <ul style="list-style-type: none"> • The path to the Reference Files • The path to a specific Reference File • A path with a wildcard pattern that matches the Reference File names (e.g. C:\Reference Files*.csv) <p>The folder can either contain a single reference file (which is continually appended with the Change File locations) or multiple reference files.</p> <p>For information on when Reference Files should be used, see <i>Reference files (page 197)</i>.</p>
Change File path is preceded by table name	<p>Select this option if each of the Change File paths in the reference files is preceded by a table name.</p> <div data-bbox="667 907 1390 1079" style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> <i>Selecting this option will disable the Table name check box in the Header columns ordinal position section.</i></p> </div>
Header column order	<p>Specify the position of each header column in your Change Files. Apart from the data columns which must be positioned after the header columns, the columns can be positioned in any order.</p> <p>So, for example, if your Change Files looked liked this:</p> <pre>DELETE, table1, timestamp, user1, dog, cat, bird</pre> <p>Then, the column positions would be set as follows:</p> <p>Operations are in column: 1</p> <p>Table names are in column: 2</p> <p>Timestamps are in column: 3</p> <p>User names are in column: 4</p> <p>Start data from column: 5</p> <p>For more information on the columns, see <i>Change files (page 198)</i>.</p> <p>See also <i>Changing the ordinal positions of the user name and timestamp columns (page 203)</i>.</p>

Option	Description
Start data from column:	Specify which column to start the actual data from. Note that data columns must be positioned after header columns. See the example in the description of the Header column order field above.

Changing the ordinal positions of the user name and timestamp columns

To change the ordinal positions of the "User name" and "Timestamp" columns, you first need to define two transformations (one for each column) that add the \$AR_H_TIMESTAMP and \$AR_H_USER header columns to the Change Files.

To add the columns to a single table (i.e. Change File):

1. Open the **Table Settings** window as described in *Defining transformations for a single table/view (page 1084)*.
2. Select the **Transform** tab and then click **Add Column**.
A new row will be added to the **Output** table.
3. Click the *fx* button at the end of the row (in the **Expression** column).
This will open the **Expression Builder** Window.
4. Select the column from the **Header Columns** tab and then click **OK**.
5. Specify a name for the column. For example, if you selected the \$AR_H_TIMESTAMP header column, specify "Timestamp".
6. Repeat Steps 1-5 to add the other header column.
7. Click **OK** to close the **Table Settings** window.

To add the header column to all tables (i.e. Change Files):

1. Open the **Global Transformation Rules** window as described in *Starting the Global Transformation Rules wizard (page 1107)*.
2. Click **Next** twice until you get to the **How to transform** screen.
3. Click the Browse button next to the **Computation expression** field.
This will open the **Expression Builder** Window.
4. Select the column from the **Header Columns** tab and then click **OK**.
5. Specify a name for the column. For example, if you selected the \$AR_H_TIMESTAMP header column, specify "Timestamp".
6. Click **Finish**.
7. Repeat steps 1-6 to add the other header column.



To determine if you are connected to the endpoint you want to use or if the connection information you entered is correct, click **Test Connection**.

If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

Defining tables and full load data

In the **Tables** tab, you specify the location of your Full Load data files and define how the source tables will be created. During Full Load, Replicate will copy the source tables to the target endpoint.

To define a table:

1. In the **Tables** tab, click **New Table**.
The **New Source Table** window opens.
2. In the **Table name** field, specify the name of the table.
3. In the **Location of full load data file(s)** field, specify location of the delimited text files that contain your full load data. Wildcard characters are supported.
4. To add a field, click **Add Field**. Then, click the default column name to edit it.
5. To specify a data type, select the row and then select a type from the drop-down list in the **Type** column.
6. When the **CLOB**, **NCLOB**, **STRING**, or **WSTRING** data types are selected, you can also set a data subtype. Select either **JSON** or **XML** from the drop-down list in the **Subtype** column. Make sure that the new data in the column will be compatible with the selected subtype. The default is **Regular**, which means that the selected data type will be used without a subtype.
7. (Optional) Click in the **Key** column to add or remove a Unique Index to/from a column.
8. To create the table, click **OK**.
The table will be added to the list of tables in the **Tables** tab.
See also the example for *Creating a Table* (page 205).

To edit a field:

- Double-click the column in the **New/Edit Source Table** dialog box and then edit the values as described above.

To delete a field:

- Select the column in the **New/Edit Source Table** dialog box and then click the **Delete Field** button.

To change the position of a field:

- Select the field and then click the **Up/Down** and **Move to Top/Move to Bottom** buttons as required.

To edit a table:

9 Adding and managing source endpoints

- In the **Tables** tab, either double-click the table or select the table and then click the **Edit** button. Edit the table as described above.

To delete a table:

- In the **Tables** tab, select the table and then click the **Delete** button.

Creating a Table

The source table definition must match the column data values in the Full Load file(s). So, for example, if the Full Load data file contains the following delimited columns:

22,January,2014,male,5463565

12,May,2011,female,3236776

9,March,2009,male,9648675

30,June,2002,female,3458795



Boolean values must be expressed as the digits 1 (TRUE) or 0 (FALSE) and not TRUE or FALSE.

Then the table definitions would look something like this:

Edit Source Table

×

Table name:

Location of full load data file(s):

+ Add Field
🗑 Delete Field

⌵
⌵
⬆
⬆

Search 🔍

Key	Name	Type
	Day	NUMERIC(18,0)
	Month	STRING(50)
	Year	NUMERIC(18,0)
	Gender	STRING(50)
<input checked="" type="checkbox"/>	<input style="width: 90%; border: 1px solid #ccc;" type="text" value="ID"/>	NUMERIC(18,0) ⌵

5 Field(s)



OK

Cancel

Setting advanced options

In the **Advanced** tab, the following options are available.

Advanced tab settings

Option	Description
File preprocessing command	<p>The source file must be in delimited text format. If your source files (Full Load and/or Change Files) are not in delimited text format, you can use your own program to convert them to the required format.</p> <p>Before the Full Load task starts, the program will be executed on Location of full load data file(s). Then, during Change Processing, it will be executed on the files specified in the Change File location.</p> <p>The command should be specified as in the following example:</p> <pre>c:\temp\files\convertfile.exe</pre> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p> <i>The path is only necessary if the conversion program's location is not defined in the "Path" system variable</i></p> </div> <p>For a detailed explanation, see <i>Setting up a File Pre-processing Command (page 207)</i>.</p>
Check for changes every	Specify how often to check the Change Files for updates.
Change Processing Cleanup	<p>Select one of the following cleanup options to determine what Replicate should do with the processed Change Files/Reference Files:</p> <ul style="list-style-type: none"> • Do nothing - to leave the file(s) in the original location. • Delete files - to delete the file(s) from the disk. • Archive files to folder - to archive the file(s) to the specified location. <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p> <i>In the case of Reference Files, the Delete files and Archive files to folder operations will only be performed if there are multiple reference files.</i></p> </div>

Setting up a File Pre-processing Command

Your program will be executed on each of the input files, and should accept the following parameters:

- The location of the input file(s) (as specified in the [Location of full load data file\(s\)](#) and/or the [Change File location](#))

- The full path of the output file

The output file(s) will be written to the following location:

```
PRODUCT_INSTALLATION\data\tasks\TASK_NAME\trans_files\OUTPUT_FILE_NAME
```

(s)

Where `OUTPUT_FILE_NAME` (s) are determined by your program.

Example:

Let's assume the following:

- The following files reside in the [Location of full load data file\(s\)](#):
c:\temp\fl\table1.csv,
c:\temp\fl\table2.csv
- The following files reside in the [Change File location](#):
c:\temp\cdc\table1.txt
c:\temp\cdc\table2.txt
- The following program is specified in the **File preprocessing command** field:
c:\temp\convertfile.exe
- Both the Full Load and Apply Changes replication options are enabled.

In such a scenario, Replicate will execute the following commands in sequence:

Before Full Load Starts:

```
c:\temp\convertfile.exe c:\temp\fl\table1.csv <PRODUCT_
INSTALLATION>\data\tasks\TASK_NAME\trans_files\table1.csv
```

```
c:\temp\convertfile.exe c:\temp\fl\table2.csv <PRODUCT_
INSTALLATION>\data\tasks\TASK_NAME\trans_files\table2.csv
```

During Change Processing:

```
c:\temp\convertfile.exe c:\temp\cdc\table1.txt <PRODUCT_
INSTALLATION>\data\tasks\TASK_NAME\trans_files\table1.txt
```

```
c:\temp\convertfile.exe c:\temp\cdc\table2.txt <PRODUCT_
INSTALLATION>\data\tasks\TASK_NAME\trans_files\table2.txt
```

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

9.7 Using Google Cloud AlloyDB for PostgreSQL as a source

This section describes how to set up the Google Cloud AlloyDB for PostgreSQL source endpoint in a replication task.

In this section:

- *Prerequisites (page 212)*
- *Limitations and considerations (page 456)*
- *Supported data types (page 214)*
- *Homogeneous replication (page 218)*
- *Setting general connection properties (page 220)*
- *Setting advanced connection properties (page 221)*
- *Removing Replicate artifacts from the database (page 223)*

Prerequisites

The following section describes the prerequisites for working with the Google Cloud AlloyDB for PostgreSQL source endpoint.

Drivers and libraries

Replicate for Windows

Install PostgreSQL ODBC Driver `psqlodbc_13_xx_xxxx-x64` on the Replicate machine.



Make sure to add the `psql.exe` path (for example, "`C:\Program Files\PostgreSQL\13xx\bin`") to the system `PATH`.

Replicate for Linux

Download PostgreSQL 13 ODBC driver, and client programs and libraries.

- **Example PostgreSQL ODBC file name:** `postgresql13-odbc-13.02.0000-1PGDG.rhel8.x86_64.rpm`
- **PostgreSQL client programs and libraries file name:** `postgresql13-13.8-1PGDG.rhel<version>.x86_64.rpm`

Then, open a shell prompt and do the following:

1. Stop the Replicate service:
`./areplicate stop`
2. Optionally, confirm that the service has stopped:
`./areplicate status`
3. Install PostgreSQL 13 client programs and libraries.
4. Install the ODBC driver for PostgreSQL 13.
5. Change the working directory to `<Replicate-Install-Dir>/bin`.
6. Copy the driver location to the `site_arep_login.sh` file:
`echo "export LD_LIBRARY_PATH=\$LD_LIBRARY_PATH:/usr/lib:/usr/lib64:/usr/pgsql-13/lib" >> site_arep_login.sh`
This will add the driver to "LD_LIBRARY_PATH" and update the driver location in the **site_arep_login.sh** file.
7. Optionally, confirm that the driver location was copied:
`cat site_arep_login.sh`
8. Makes sure that the `/etc/odbcinst.ini` file contains an entry for PostgreSQL, as in the following example:

```
[PostgreSQL Unicode(x64)]
Description = PostgreSQL ODBC driver
Driver = /usr/pgsql-13/lib/psqlodbc.so
Setup = /usr/pgsql-13/lib/psqlodbcw.so
UsageCount = 1
```
9. Start the Replicate service:
`./areplicate start`
10. Optionally, confirm that the service has started:
`./areplicate status`

Prerequisites for capturing changes

To be able to capture changes from the database, you need to set up logical replication.

To do this:

1. On Google Cloud platform, set the following flags in the AlloyDB settings:
 - `alloydb.logical_decoding=on`
 - **max_replication_slots**: Every task creates a new slot; this parameter configures the amount of simultaneous connections that can be established to the database. If you encounter performance issues, increasing this value might help.
 - **max_wal_senders**: The number of tasks that can run in parallel. If you encounter performance issues, increasing this value might help.

For more information on these parameters, see <https://www.postgresql.org/docs/current/runtime-config-replication.html>

2. On the database, alter the Replicate user to be able to work with replication:

```
ALTER USER replicate-user WITH REPLICATION;
```

Where `replicate-user` is the user specified in the endpoint connection settings.

Required permissions

The user specified in the endpoint connection settings must be granted the following permissions on the Google Cloud AlloyDB for PostgreSQL database:

- **For Full Load tasks**: Standard SELECT
- **For CDC tasks**: No special permissions are required

Limitations and considerations

You should be aware of the following limitations and considerations when using a PostgreSQL source:

- The database name cannot include a semi-colon (;).
- Capturing changes while the database is in Hot Standby mode is not supported.
- To capture changes from a secondary database, read/write privileges must be configured on the database. Due to the write permission requirement, the PostgreSQL read replica (in a High Availability environment) is not supported.
- Both the source table and the corresponding target table must have an identical Primary Key. In the event that one of the tables does not have a Primary Key, the result of DELETE and UPDATE record operations will be unpredictable.
- The “Start Process Changes from Timestamp” run option is not supported.
- Replication of the Before Image is not supported.
- Replication of multiple tables with the same name but a different case (e.g. `table1`, `TABLE1` and `Table1`) may cause unpredictable behavior and is therefore not supported.
- Change processing of [CREATE | ALTER | DROP] table DDLs are supported unless they are held in an inner function/procedure body block or in other nested constructs.

For example, the following change will not be captured:

```
CREATE OR REPLACE FUNCTION attu.create_distributors1() RETURNS void
LANGUAGE plpgsql
```

```
AS $$
BEGIN
  create table attu.distributors1(did serial PRIMARY KEY,name varchar(40) NOT NULL);
END;
$$;
```

- Change processing of TRUNCATE operations is not supported.
- Deferred Constraints are not supported.
- **Partitioned tables**
 - CDC is not supported.
- When using the *Parallel Load (page 1098)* feature, table segmentation according to partitions or sub-partitions is not supported.
- When using [filters](#) that do not contain a Primary Key, DELETE operations will not be captured.
- The default configuration of Replica identity is supported only.
For more information about Replica identity, see <https://www.postgresql.org/docs/10/logical-replication-publication.html>
- A DDL operation that occurs during Full Load, but that was not preceded by a DML operation will not be captured.
- PostGIS datatypes are not supported.
- Unique Indexes with CASE expressions are not supported.
- CDC is supported with the primary instance only
- The SQL MERGE command introduced in PostgreSQL 15.x has not been certified.

Supported data types

The following table shows the PostgreSQL source data types that are supported when using Qlik Replicate and the default mapping to the Qlik Replicate data types.


When replicating from Google Cloud AlloyDB for PostgreSQL to a PostgreSQL-based target, source and target data types are identical for all supported PostgreSQL versions, except for the data types listed in *Data type considerations and exceptions (page 219)*



For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

Data types



PostgreSQL	Qlik Replicate Data Types
INTEGER	INT4
SMALLINT	INT2
BIGINT	INT8
NUMERIC	NUMERIC(28,6)

9 Adding and managing source endpoints

PostgreSQL	Qlik Replicate Data Types
NUMERIC(p,s)	<p>If precision is ≤ 38, then: NUMERIC(p,s)</p> <p>If precision is ≥ 39, then: STRING</p>
DECIMAL	NUMERIC(28,6)
DECIMAL(p,s)	<p>If precision is ≤ 38, then: NUMERIC(p,s)</p> <p>If precision is ≥ 39, then: STRING</p>
REAL	REAL4
DOUBLE	REAL8
SMALLSERIAL	INT2
SERIAL	INT4
BIGSERIAL	INT8
MONEY	<p>NUMERIC(38,4)</p> <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;"> <p> <i>The MONEY data type is mapped to FLOAT in Microsoft SQL Server.</i></p> </div>
CHAR	WSTRING (1)
CHAR(n)	WSTRING (n)

PostgreSQL	Qlik Replicate Data Types
VARCHAR(n) <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> <i>VARCHAR without a length (n) is not recognized as a valid data type by target endpoints. Consequently, if a source column data type is set to VARCHAR without an explicit length, Replicate will set a default length of 8000 bytes.</i></p> <p><i>You can change the default by setting the following internal parameter to the required length:</i></p> <p><i><code>unboundedVarcharMaxSize</code></i></p> <p><i>See also: Internal Parameters.</i></p> </div>	WSTRING (n)
TEXT	NCLOB
BYTEA	BLOB
TIMESTAMP	DATETIME
TIMESTAMP (z)	DATETIME
DATE	DATE
<div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> <i>Replicate only supports ISO formatted textual DATE formats (the default). If other formats are used, an error will be generated. You can change the date format in the <code>postgresql.conf</code> file or using the <code>PGDATESTYLE</code> environment variable. You can also change the date format at database level.</i></p> </div>	
TIME	TIME
TIME (z)	TIME
INTERVAL	STRING (128) - 1 YEAR, 2 MONTHS, 3 DAYS, 4 HOURS, 5 MINUTES, 6 SECONDS
BOOLEAN	STRING (5)
ENUM	STRING (64)
CIDR	STRING (50)
INET	STRING (50)
MACADDR	STRING (18)
BIT (n)	STRING (n)

9 Adding and managing source endpoints

PostgreSQL	Qlik Replicate Data Types
BIT VARYING (n)	STRING (n)
UUID	STRING
TSVECTOR	CLOB
TSQUERY	CLOB
XML	CLOB
POINT	STRING (255) "(x,y)"
LINE	STRING (255) "(x,y,z)"
LSEG	STRING (255) "((x1,y1),(x2,y2))"
BOX	STRING (255) "((x1,y1),(x2,y2))"
PATH	CLOB "((x1,y1),(xn,yn))"
POLYGON	CLOB "((x1,y1),(xn,yn))"
CIRCLE	STRING (255) "(x,y),r"
JSON	NCLOB
<div style="border: 1px solid gray; padding: 10px;"> <p> Reading and writing JSON data is performed using wide string (where each character occupies 2 bytes). Therefore, when replicating JSON columns to a PostgreSQL target, set the limited lob size to twice the maximum JSON length. So, for example, if the JSON data is 4 KB, set the limited lob size value to 8 KB.</p> </div>	
JSONB	NCLOB
<div style="border: 1px solid gray; padding: 10px;"> <p> Reading and writing JSONB data is performed using wide string (where each character occupies 2 bytes). Therefore, when replicating JSONB columns to a PostgreSQL target, set the limited lob size to twice the maximum JSONB length. So, for example, if the JSONB data is 4 KB, set the limited lob size value to 8 KB.</p> </div>	
ARRAY	NCLOB
COMPOSITE	NCLOB
INT4RANGE	STRING (255)
INT8RANGE	STRING (255)
NUMRANGE	STRING (255)

PostgreSQL	Qlik Replicate Data Types
TSRANGE	STRING (255)
INT4MULTIRANGE	STRING (255)
INT8MULTIRANGE	STRING (255)
NUMMULTIRANGE	STRING (255)
TSMULTIRANGE	STRING (255)
CHARACTER VARYING	If length is specified: WSTRING (LENGTH) If no length is specified: WSTRING (8000)
TINTERVAL	WSTRING(255)



The PostGIS data type is not supported.

Homogeneous replication

The following section describes how Replicate handles replication from Google Cloud AlloyDB for PostgreSQL to the following targets: PostgreSQL, Microsoft Azure Database for PostgreSQL, Google Cloud SQL for PostgreSQL, and Google Cloud AlloyDB for PostgreSQL.

In homogeneous replication, source column and table collations will be replicated to the target as described in *Column and table collation (page 219)*.



In homogeneous replication, the source data first passes through the Qlik Replicate data type and is therefore subject to any limitations of that type.

*For information on Replicate data types and their limitations (where relevant), see *Replicate data types (page 81)*.*

*For information on which Replicate data types the source data passes through when replicating from Google Cloud AlloyDB for PostgreSQL, see *Supported data types (page 214)*.*

Limitations and considerations

Homogeneous replication is not supported with the following Replicate configurations or operations:


- Apply Changes only tasks
- Stopping and resuming a task
- Starting a task from a timestamp

- When the **If target table already exists** in the task settings' **Full Load Settings** tab is set to **Do nothing**

Data type considerations and exceptions

When replicating from Google Cloud AlloyDB for PostgreSQL to a PostgreSQL-based target, source and target data types will be identical for all supported PostgreSQL versions, with the following exceptions:

Data type exceptions in homogenous replication

Google Cloud AlloyDB for PostgreSQL Source	PostgreSQL Target
ENUM	STRING
COMPOSITE	STRING
NUMERIC	NUMERIC (28,6)
NUMERIC (p,s)	NUMERIC
A numeric array with precision and scale.	A numeric array without precision and scale.
TINTERVAL	 <p>As PostgreSQL have discontinued support for the TINTERVAL data type, tables with this data type will not be created on the target if the target database version does not support it.</p>
INT4MULTIRANGE	STRING (255)
INT8MULTIRANGE	STRING (255)
NUMMULTIRANGE	STRING (255)
TSMULTIRANGE	STRING (255)

See also the note about JSON data type replication in *Supported data types (page 214)*.

Column and table collation

When replicating from Google Cloud AlloyDB for PostgreSQL to a PostgreSQL-based target, column and table collations will be replicated to the target.



To support collation replication, the DBA must ensure that the collations defined for the Google Cloud AlloyDB for PostgreSQL source database are the same as those defined for the PostgreSQL-based target database.

Non-nullable columns and primary/unique index names

Non-nullable columns and Primary/Unique Index names are preserved during homogeneous replication.

Replication of Partitioned Tables

In order to replicate partitioned tables from Google Cloud AlloyDB for PostgreSQL to a PostgreSQL-based target, you need to do the following:

1. Manually create the parent and child tables in the target database.
2. Define a separate task to replicate to those tables.
3. Set the **Truncate before loading** option in the *Full Load Settings* (page 1171).

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a Google Cloud AlloyDB for PostgreSQL source endpoint source to Qlik Replicate:

1. In the Qlik Replicate console, click **Manage Endpoint Connections** to open the **Manage Endpoint Connections** dialog box. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints* (page 114).
2. In the **Name** field, type a name for your PostgreSQL database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the PostgreSQL database. This is optional.
4. Select **Source** as the database **role**.
5. Select **Google Cloud AlloyDB for PostgreSQL** as the database **Type**.



*When this endpoint is used as a duplicated source in a Log Stream Staging setup, select the **Read changes from log stream** check box and then select the relevant Log Stream Staging task from the drop-down list.*

*For information on setting up and managing Log Stream Staging tasks, see *Using the Log Stream* (page 1072).*

6. Type the **Server** name. This is the name or IP address of the computer with the PostgreSQL database that you want to access.
7. Optionally, change the default port (5432).
8. Enter the PostgreSQL database authentication information (**User name**, **Password**) of an authorized PostgreSQL user.



Consider the following:

- This information is case sensitive.
- To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**. If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box. To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

9. Type the **Database** name or browse for a source database.

SSL authentication options

In the **Security** section, you can configure Replicate to connect to the PostgreSQL database using SSL.

- **SSL Mode:** Select one of the following:
 - **Disable** - Connect with a surname and password only.
 - **Allow** - Establish an encrypted connection if requested by the server.
 - **Prefer** - Establishes an encrypted connection if the server supports encrypted connections, falling back to an unencrypted connection if an encrypted connection cannot be established.
 - **Require** - Establishes an encrypted connection if the server supports encrypted connections. The connection attempt fails if an encrypted connection cannot be established.
 - **Verify CA** - Similar to **Required**, but also verifies the server Certificate Authority (CA) certificate against the configured CA certificates. The connection attempt fails if no valid matching CA certificates are found.
 - **Verify Full** - Similar to **Verify CA**, but also performs host name identity verification by checking the host name the client (i.e. Replicate) uses for connecting to the server against the identity in the certificate that the server sends to the client. The client checks whether the host name that it uses for connecting matches the Common Name value in the server certificate. The connection fails if there is a mismatch.
- **Client certificate path** - The path to the client certificate requested by the server.
- **Secret key path** - The path to the client private key file in PEM format.
- **CA path** - The path to the Certificate Authority (CA) that issued the client certificate file in PEM format.
- **CRL path** - The path to the CRL certificate. This file contains certificates revoked by certificate authorities. If the server certificate appears in this list, the connection will fail.
- **SSL compression** - Select this option to compress the data before it is encrypted.

Setting advanced connection properties

In the **Advanced** tab, you can set the following properties:

- **Capture DDL events:** When this option is selected, the following actions occur:
 - Operational artifacts are created (by Replicate) in the database when the task starts. In order to capture DDL events, Qlik Replicate creates various artifacts in the PostgreSQL database when the task starts. You can later remove these artifacts as described in *Removing Replicate artifacts from the database (page 223)*.
 - Streamed DDL events are captured.
- **Create DDL artifacts in schema:** The schema in which the operational DDL database artifacts will be created. The default value is "Public".
- **WAL heartbeat** - An Apply Changes task that is running but not capturing changes (due to source table inactivity) will continue to occupy the LSN position in its replication slot, thereby preventing truncation of the WAL. Since the WAL is a server-wide resource used by all PostgreSQL processes, it may grow extremely large if no changes are captured for an extended period. To prevent this from happening, enable the "WAL heartbeat" option. When this option is enabled, the PostgreSQL source endpoint mimics task activity by periodically committing pseudo transactions (i.e. "Heartbeats") to the heartbeat table, thereby advancing the task slot's LSN position.
 - **Schema for heartbeat artifacts:** The schema in which the WAL heartbeat table (attrep_wal_heartbeat) will be created. The default value is "public".
 - **Heartbeat frequency (minutes):** The frequency with which to commit transactions to the heartbeat table.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

Removing Replicate artifacts from the database

In order to capture DDLs, Qlik Replicate creates various artifacts in the PostgreSQL database when the task starts. When the task completes, you may wish to remove these artifacts.

To remove the artifacts, execute the following statements (in the order they appear below), where `public` is the default schema in which the artifacts were created:



The event trigger does not belong to a specific schema.

```
drop event trigger attrep_intercept_ddl;  
drop function public.attrep_intercept_ddl()  
drop table public.attrep_ddl_audit
```

9.8 Using Google Cloud SQL for MySQL as a source

This section describes how to set up and use Google Cloud SQL for MySQL as a source in a replication task.

In this section:

- *Prerequisites (page 225)*
- *Limitations and considerations (page 226)*
- *Permissions (page 227)*
- *Data types (page 227)*
- *Setting general connection properties (page 231)*
- *Selecting a schema (page 233)*
- *Setting advanced connection properties (page 233)*

Prerequisites

Before using Google Cloud SQL for MySQL as an endpoint in a Replicate task, make sure the prerequisites outlined below have been met.

Driver prerequisites

Replicate on Windows

Install MySQL ODBC Unicode Driver 64-bit version 8.0.32 on the Replicate machine.

Replicate on Linux

1. Stop the Replicate service:

```
./areplicate stop
```
2. Optionally, confirm that the service has stopped:

```
./areplicate status
```
3. Install MySQL ODBC Unicode Driver 64-bit version 8.0.32 on the Replicate machine.
4. Change the working directory to *<Replicate-Install-Dir>/bin*.
5. Copy the driver location to the `site_arep_login.sh` file:

```
echo "export LD_LIBRARY_PATH=\$LD_LIBRARY_PATH:/usr/lib64" >> site_arep_login.sh
```

This will add the driver to "LD_LIBRARY_PATH" and update the driver location in the **site_arep_login.sh** file.
6. Optionally, confirm that the driver location was copied:

```
cat site_arep_login.sh
```
7. Make sure that the `/etc/odbcinst.ini` file contains an entry for MySQL, as in the following example:

```
[MySQL ODBC 8.0 Unicode Driver]  
Driver = /usr/lib64/libmyodbc8w.so  
UsageCount = 1
```
8. Start the Replicate service:

```
./areplicate start
```
9. Optionally confirm that the service has started:

```
./areplicate status
```

Google Cloud SQL for MySQL replication

Replication enables data from one Google Cloud SQL for MySQL database server (the primary server) to be copied to one or more Google Cloud SQL for MySQL database servers (the secondary servers).

The Replicate Google Cloud SQL for MySQL source endpoint can be configured to replicate data from either a the primary server or a secondary server.

To replicate changes from a secondary server (CDC), the binary logging parameter `log_slave_updates` needs to be set to `true` (1).

Verifying that Google Cloud SQL for MySQL is set up for CDC (Change Data Capture)

Google Cloud SQL for MySQL enables binary logging (i.e. CDC) by default. To check that binary logging is indeed enabled, make sure that the Google Cloud SQL for MySQL database `binlog_row_image` parameter is set to `full`.

Replicating 4-byte UTF8 emojis

Replication of 4-byte UTF8 emojis to certain targets requires the following preparation:

- **Microsoft SQL Server Target:** Transform the emojis from `WSTRING (n)` to `WSTRING (n*2)`.
- **Amazon Redshift Target:** Transform the emojis from `WSTRING (n)` to `WSTRING (n*2)`.

For information on defining transformations, see *Using the Transform tab (page 1086)*.

Limitations and considerations

The following limitations apply:

- The following DDLs are not supported:
 - All partition DDLs
 - Drop Table
 - Rename Table
- Changes that are older than one year will not be captured.
- Using the `alter table <table_name> add column <column_name>` statement to add columns to the beginning or to the middle of a table is not supported. When the `alter table <table_name> add column <column_name>` statement is used to add a column to the middle of a table, the column will be added to the end of the table instead.
- The **AR_H_USER** header column is currently not supported. For information on using header columns, see *Headers (page 1143)*.
- If an Google Cloud SQL for MySQL table contains LOBs and the task's **Replicate Lob columns** option is disabled, the table will be replicated without the LOB columns. Note that this only applies to MEDIUMBLOB, LONGBLOB, MEDIUMTEXT and LONGTEXT columns. This limitation does not apply to BLOB, TINYBLOB, TEXT and TINYTEXT columns.
- If the Google Cloud SQL for MySQL database is stopped during Full Load, the Full Load will end successfully, but the tables on the target may have less rows than the source tables. If this should happen, either restart the task or reload the tables with the missing rows.
- A Unique Index/Primary Key created on only part of the column data is not supported. If such a Unique Index/Primary Key is encountered, it will be created on the entire target column. The following is an example of a statement that creates a Unique Index/Primary Key using only part of the column data:

```
CREATE INDEX partial_name ON customer (name(10));
```
- Changes to computed (virtual) columns will not be captured.
- Replicate supports zero dates only if both the source and target MySQL-based databases are configured with the same "Strict Mode" (On or Off). If this is not the case, zero dates can be transformed to non-zero dates [using the transformation feature](#).
- The Transaction ID of XA transactions will not be displayed in the replication ChangeTable/Audit Table **transaction_id** header.
- Binary log transaction compression is not supported.
- The MySQL primary-primary configuration is not supported.
- Failover of cluster nodes is not supported.

- Capturing changes from an encrypted binlog is not supported.
- Due to an issue with MySQL ODBC Driver 8.0.027 or later, empty TEXT columns may not be replicated correctly during Full Load. For example, if one table row contains a TEXT column with a value and the same column in the next row contains an empty value (but not NULL), both rows will display the value of the first row on the target.

Permissions

The user specified in the endpoint connection settings must have the following permissions:

Full Load tasks

The ReplicationAdmin role with the SELECT privilege for the designated source tables.

Apply Changes tasks

The ReplicationAdmin role with the REPLICATION CLIENT and REPLICATION SLAVE privileges.

Checking for open XA transactions

When the task is initialized, Replicate checks whether there are any open XA transactions (transactions that use a two phase commit protocol). To allow this query to be performed, the following permission needs to be granted:

```
grant XA_RECOVER_ADMIN on *.* to 'userName'@'%';
```

Where `userName` is the user specified in the endpoint settings.

If this permission is not granted, the following error will be shown when you run the task:

```
Failed to check for open XA transactions. Verify that the user specified in the endpoint settings has the XA_RECOVER_ADMIN permission. Capturing changes while there are open XA transactions might lead to loss of data. If there are no XA transactions in this database, add the internal parameter IgnoreOpenXaTransactionsCheck to skip the XA transactions check.
```

Data types

The following table shows the Google Cloud SQL for MySQL database source data types that are supported when using Qlik Replicate and the default mapping to Qlik Replicate data types. When replicating to a MySQL target, the source and target data types are the same, apart from the exceptions described in *Homogeneous Replication* (page 230).




For information on how to view the data type that is mapped in the target, see the section for the target database you are using.

For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).

Supported data types

Google Cloud SQL for MySQL source data types	Qlik Replicate data types
INT	INT4
BIGINT	INT8

9 Adding and managing source endpoints

Google Cloud SQL for MySQL source data types	Qlik Replicate data types
MEDIUMINT	INT4
TINYINT	INT1
SMALLINT	INT2
UNSIGNED TINYINT	UINT1
UNSIGNED SMALLINT	UINT2
UNSIGNED MEDIUMINT	UINT4
UNSIGNED INT	UINT4
UNSIGNED BIGINT	UINT8
DECIMAL (10)	NUMERIC (10,0)
BINARY	BYTES (1)
BIT	BOOLEAN
BIT (64)	BYTES (8)
BLOB	BYTES (65535)
LOB	BLOB
MEDIUMBLOB	BLOB
TINYBLOB	BYTES (255)
DATE	DATE
DATETIME	DATETIME
<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <p> <i>DATETIME without a parenthetical value is replicated without milliseconds, whereas DATETIME with a value of 1-5 - e.g. DATETIME(5) - is replicated with milliseconds.</i></p> </div> <div style="border: 1px solid #ccc; padding: 5px;"> <p> <i>When replicating a DATETIME column, the time remains the same on the target (i.e. it is not converted to UTC).</i></p> </div>	
TIME	STRING
TIMESTAMP	DATETIME
<div style="border: 1px solid #ccc; padding: 5px;"> <p> <i>When replicating a TIMESTAMP column, the time is converted to UTC on the target.</i></p> </div>	
YEAR	INT2

9 Adding and managing source endpoints

Google Cloud SQL for MySQL source data types	Qlik Replicate data types
DOUBLE	REAL8
FLOAT If the FLOAT values are not in the range specified below, use a transformation to map FLOAT to STRING. For an explanation of how to do this, see <i>Using the Transform tab (page 1086)</i> . Supported FLOAT range: - 1.79E+308 to -2.23E-308, 0 and 2.23E-308 to 1.79E+308	REAL (DOUBLE)
*VARCHAR (45)	WSTRING (45)
*VARCHAR (2000)	WSTRING (2000)
*VARCHAR (4000)	WSTRING (4000)
VARBINARY (4000)	BYTES (4000)
VARBINARY (2000)	BYTES (2000)
*CHAR	WSTRING
*TEXT	WSTRING (65535)
*LONGTEXT	NCLOB
*MEDIUMTEXT	NCLOB
*TINYTEXT	WSTRING (255)
GEOMETRY	BLOB
POINT	BLOB
LINESTRING	BLOB
POLYGON	BLOB
MULTIPOINT	BLOB
MULTILINESTRING	BLOB
MULTIPOLYGON	BLOB
GEOMETRYCOLLECTION	BLOB
ENUM	WSTRING (Length) Where "Length" is the longest value in the ENUM.

Google Cloud SQL for MySQL source data types	Qlik Replicate data types
SET	WSTRING (Length) Where "Length" is the total of all values in the SET, including commas.
JSON	CLOB



If the DATETIME and TIMESTAMP data types are specified with a “zero” value (i.e. 0000-00-00), you need to make sure that the target database in the replication task supports "zero" values for the DATETIME and TIMESTAMP data types. If they are not supported, you can use a transformation to specify a supported value (e.g. 1970.) Otherwise, they will be recorded as null on the target.

Homogeneous Replication

The following section describes how Replicate handles replication between an Google Cloud SQL for MySQL source and the following targets: MySQL, Microsoft Azure Database for MySQL, and Google Cloud SQL for MySQL.



- In homogeneous replication, the source data first passes through the Qlik Replicate data type and is therefore subject to any limitations of that type.
For information on Replicate data types and their limitations (where relevant), see Replicate data types (page 81).
For information on which Replicate data types the source data passes through when replicating from Google Cloud SQL for MySQL, see the Google Cloud SQL for MySQL to Qlik Replicate data types mapping table described earlier.*
- When replicating a TIMESTAMP column, the time is converted to UTC on the target.*

Limitations and considerations

Homogeneous replication is not supported with the following Replicate configurations or operations:

- Apply Changes only tasks
- Stopping and resuming a task
- Starting a task from a timestamp
- When the **If target table already exists** in the task settings' **Full Load Settings** tab is set to **Do nothing**

Data types

The source and target data types will be identical with the following exceptions:

MySQL source and MySQL-based target data types

Google Cloud SQL for MySQL source data types	MySQL-based target data types
JSON	LONGTEXT



When using data types from other database engines, the data type on the target will be the corresponding MySQL data type. For more information on using non-native data types, refer to the MySQL online Help.

Collation

Table and column collations will be replicated to the target. Collatable data types are indicated by an asterisk (*) in the *Supported data types* (page 227) table.

To support collation replication, the DBA must ensure that the collations defined for the source Google Cloud SQL for MySQL database are the same as those defined for the MySQL-based target database.

Non-nullable columns and Primary/Unique index names

Non-nullable columns and Primary/Unique Index names are preserved during homogeneous replication.

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.



You can also use Google Cloud SQL for MySQL files as a source. For more information, see *Using the Qlik Replicate file channel* (page 1061).

To add an Google Cloud SQL for MySQL source endpoint to Qlik Replicate:

1. In the Qlik Replicate Console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints* (page 114).
2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Google Cloud SQL for MySQL database. This is optional.
4. Select **Source** as the endpoint **role**.
5. From the **Type** drop-down list, select **Google Cloud SQL for MySQL**.



When this endpoint is used as a duplicated source in a Log Stream Staging setup, select the **Read changes from log stream** check box and then select the relevant Log Stream Staging task from the drop-down list.

For information on setting up and managing Log Stream Staging tasks, see *Using the Log Stream* (page 1072).

6. In the **Server** field, enter the host name or IP address of the computer on which the Google Cloud SQL for MySQL database is installed.
7. Optionally, change the default port (3306).
8. Type the Google Cloud SQL for MySQL authentication information (**Username, Password**) for the authorized user for this Google Cloud SQL for MySQL database. If you do not know this information, see your Google Cloud SQL for MySQL database Administrator (DBA).



- The **User Name, Password** are case sensitive.
- Passwords containing curly brackets are not supported i.e. {}.



Make sure that the Google Cloud SQL for MySQL user entered in the Google Cloud SQL for MySQL Authentication section has the correct access privileges. For information on how to provide the required privileges, see *Permissions* (page 227).

SSL authentication options

In the **Security** section, you can configure Replicate to connect to the Google Cloud SQL for MySQL database using SSL.

- **SSL Mode:** Select one of the following:
 - **Preferred** - Establishes an encrypted connection if the server supports encrypted connections, falling back to an unencrypted connection if an encrypted connection cannot be established.
 - **None** - Connect with a surname and password only.
 - **Required** - Establishes an encrypted connection if the server supports encrypted connections. The connection attempt fails if an encrypted connection cannot be established.
 - **Verify CA** - Similar to **Required**, but also verifies the server Certificate Authority (CA) certificate against the configured CA certificates. The connection attempt fails if no valid matching CA certificates are found.
 - **Verify Full** - Similar to **Verify CA**, but also performs host name identity verification by checking the host name the client (i.e. Replicate) uses for connecting to the server against the identity in the certificate that the server sends to the client. The client checks whether the host name that it uses for connecting matches the Common Name value in the server certificate. The connection fails if there is a mismatch.
- **Client certificate path** - The path to the client certificate requested by the server.
- **Private key path** - The path to the client private key file in PEM format.
- **CA path** - The path to the Certificate Authority (CA) that issued the client certificate file in PEM format.

Selecting a schema

You can choose which Google Cloud SQL for MySQL database to access. After configuring the Google Cloud SQL for MySQL source database connection settings, open the **Select Tables** dialog box (by clicking the **Table Selection** button on the right of the console) and select which schema to use from the **Schema** drop down list.

See also *Defining and managing tasks* (page 109).

Setting advanced connection properties

In the **Advanced** tab, you can set the following parameters:

- **Check binary log for new events every:** Specify how often to check the binary log for changes when the endpoints is idle.
- **Additional ODBC connection properties:** Specify any additional ODBC connection parameters that may be required.
- **Database timezone:** Select the timezone where the source database is located.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

9.9 Using Google Cloud SQL for PostgreSQL as a source

This section describes how to set up and use a Google Cloud SQL for PostgreSQL database as a source in a replication task.

In this section:

- *Prerequisites (page 235)*
- *Required permissions (page 236)*
- *Limitations and considerations (page 236)*
- *Supported data types (page 237)*
- *Homogeneous Replication (page 241)*
- *Setting general connection properties (page 243)*
- *Setting advanced connection properties (page 244)*
- *Removing Replicate artifacts from the Google Cloud SQL for PostgreSQL database (page 246)*

Prerequisites

The following section lists the prerequisites for working with Qlik Replicate and a Google Cloud SQL for PostgreSQL source.

Replicate for Windows

PostgreSQL ODBC Driver `psqlodbc_13_xx_xxxx-x64` must be installed on the Replicate machine.



Make sure that the `psql.exe` path (e.g. "`C:\Program Files\PostgreSQL\13xx\bin`") is added to the system `PATH`.

Replicate for Linux

Download PostgreSQL 13 ODBC driver, and client programs and libraries.

- **Example PostgreSQL ODBC file name:** `postgresql13-odbc-13.02.0000-1PGDG.rhel8.x86_64.rpm`
- **PostgreSQL client programs and libraries file name:** `postgresql13-13.8-1PGDG.rhel<version>.x86_64.rpm`

Then, open a shell prompt and do the following:

1. Stop the Replicate service:
`./areplicate stop`
2. Optionally, confirm that the service has stopped:
`./areplicate status`
3. Install PostgreSQL 13 client programs and libraries.
4. Install the ODBC driver for PostgreSQL 13.
5. Change the working directory to `<Replicate-Install-Dir>/bin`.
6. Copy the driver location to the `site_arep_login.sh` file:
`echo "export LD_LIBRARY_PATH=\$LD_LIBRARY_PATH:/usr/lib:/usr/lib64:/usr/pgsql-13/lib" >> site_arep_login.sh`
This will add the driver to "LD_LIBRARY_PATH" and update the driver location in the **site_arep_login.sh** file.
7. Optionally, confirm that the driver location was copied:
`cat site_arep_login.sh`
8. Makes sure that the `/etc/odbcinst.ini` file contains an entry for PostgreSQL, as in the following example:

```
[PostgreSQL Unicode(x64)]
Description = PostgreSQL ODBC driver
Driver = /usr/pgsql-13/lib/psqlodbc.so
Setup = /usr/pgsql-13/lib/psqlodbcw.so
UsageCount = 1
```
9. Start the Replicate service:
`./areplicate start`
10. Optionally, confirm that the service has started:
`./areplicate status`

CDC prerequisites

To be able to capture changes from the database, you need to set up logical replication.

To set up logical replication:

1. On Google Cloud platform, set the following flags in the PostgreSQL settings:
 - `cloudsql.logical_decoding=on`
 - `max_replication_slots`: Every connection (that is, task) creates a new slot; this parameter configures the amount of simultaneous connections that can be established to the database. If you encounter performance issues, increasing this value might help.
 - `max_wal_senders`: The number of tasks that can run in parallel. If you encounter performance issues, increasing this value might help.

For more information on these parameters, see <https://www.postgresql.org/docs/current/runtime-config-replication.html>

2. On the database, alter the user specified in the endpoint settings (`username`) to support replication:

```
ALTER USER username WITH REPLICATION;
```

Required permissions

The user specified in the endpoint connection settings must be granted the following permissions on the Google Cloud SQL for PostgreSQL database:

- **For Full Load tasks:** Standard SELECT
- **For CDC tasks:** Superuser

Limitations and considerations

You should be aware of the following limitations and considerations when using Google Cloud SQL for PostgreSQL as a source:

- The database name cannot include a semi-colon (;).
- Capturing changes while the database is in Hot Standby mode is not supported.
- To capture changes from a secondary database, read/write privileges must be configured on the database. Due to the write permission requirement, the PostgreSQL read replica (in a High Availability environment) is not supported.
For information about the read-write permission, see *Prerequisites (page 453)*
- Replication of multiple tables with the same name but a different case (e.g. `table1`, `TABLE1` and `Table1`) may cause unpredictable behavior and is therefore not supported.
- Change processing of TRUNCATE operations is not supported.
- Deferred Constraints are not supported.
- The default configuration of Replica identity is supported only.
For more information about Replica identity, see <https://www.postgresql.org/docs/10/logical-replication-publication.html>
- Unique Indexes with CASE expressions are not supported.

9 Adding and managing source endpoints

- Both the source table and the corresponding target table must have an identical Primary Key. In the event that one of the tables does not have a Primary Key, the result of DELETE and UPDATE record operations will be unpredictable.
- The “Start Process Changes from Timestamp” run option is not supported.
- Replication of the Before Image is not supported.
- Change processing of [CREATE | ALTER | DROP] table DDLs are supported unless they are held in an inner function/procedure body block or in other nested constructs.

For example, the following change will not be captured:

```
CREATE OR REPLACE FUNCTION attu.create_distributors1() RETURNS void
LANGUAGE plpgsql
AS $$
BEGIN
    create table attu.distributors1(did serial PRIMARY KEY,name varchar(40) NOT NULL);
END;
$$;
```

- **Partitioned tables**
 - CDC is not supported.
- When using the [Parallel Load](#) feature, table segmentation according to partitions or sub-partitions is not supported.
- When using [filters](#) that do not contain a Primary Key, DELETE operations will not be captured.
- A DDL operation that occurs during Full Load, but that was not preceded by a DML operation will not be captured.
- PostGIS data types are not supported.
- Forced failover is a feature of Azure Database for PostgreSQL – Flexible Server. If forced failover occurs when replicating data from Azure Database for PostgreSQL – Flexible Server, the task will stop and will need to be reloaded.
- The SQL MERGE command introduced in PostgreSQL 15.x has not been certified.

Unsupported data types



The PostGIS data type is not supported.

Supported data types


The following table shows the Google Cloud SQL for PostgreSQL source data types that are supported when using Qlik Replicate and the default mapping to the Qlik Replicate data types.



For information about data type mapping in homogeneous replication, see *Homogeneous Replication (page 241)*.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.



Google Cloud SQL for PostgreSQL data types	Qlik Replicate data types
INTEGER	INT4

9 Adding and managing source endpoints

Google Cloud SQL for PostgreSQL data types	Qlik Replicate data types
SMALLINT	INT2
BIGINT	INT8
NUMERIC	NUMERIC(28,6)
NUMERIC(p,s)	<p>If precision is ≤ 38, then:</p> <p>NUMERIC(p,s)</p> <p>If precision is > 39, then:</p> <p>STRING</p>
DECIMAL	NUMERIC(28,6)
DECIMAL(p,s)	<p>If precision is ≤ 38, then:</p> <p>NUMERIC(p,s)</p> <p>If precision is > 39, then:</p> <p>STRING</p>
REAL	REAL4
DOUBLE	REAL8
SMALLSERIAL	INT2
SERIAL	INT4
BIGSERIAL	INT8
MONEY	<p>NUMERIC(38,4)</p> <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;"> <p> <i>The MONEY data type is mapped to FLOAT in Microsoft SQL Server.</i></p> </div>
CHAR	WSTRING (1)
CHAR(n)	WSTRING (n)

Google Cloud SQL for PostgreSQL data types	Qlik Replicate data types
VARCHAR(n) <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> <i>VARCHAR without a length (n) is not recognized as a valid data type by target endpoints. Consequently, if a source column data type is set to VARCHAR without an explicit length, Replicate will set a default length of 8000 bytes.</i></p> <p><i>You can change the default by setting the following internal parameter to the required length:</i></p> <p style="margin-left: 20px;"><code>unboundedVarcharMaxSize</code></p> <p><i>See also Internal Parameters.</i></p> </div>	WSTRING (n)
TEXT	NCLOB
BYTEA	BLOB
TIMESTAMP	DATETIME
TIMESTAMP (z)	DATETIME
<div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> <i>Replicate only supports ISO formatted textual DATE formats (the default). If other formats are used, an error will be generated. You can change the date format in the Google Cloud SQL for PostgreSQL.conf file or using the PGDATESTYLE environment variable. You can also change the date format at database level.</i></p> </div>	
DATE	DATE
TIME	TIME
TIME (z)	TIME
INTERVAL	STRING (128) - 1 YEAR, 2 MONTHS, 3 DAYS, 4 HOURS, 5 MINUTES, 6 SECONDS
BOOLEAN	STRING (5)
ENUM	STRING (64)
CIDR	STRING (50)
INET	STRING (50)
MACADDR	STRING (18)

9 Adding and managing source endpoints

Google Cloud SQL for PostgreSQL data types	Qlik Replicate data types
BIT (n)	STRING (n)
BIT VARYING (n)	STRING (n)
UUID	STRING
TSVECTOR	CLOB
TSQUERY	CLOB
XML	CLOB
POINT	STRING (255) "(x,y)"
LINE	STRING (255) "(x,y,z)"
LSEG	STRING (255) "((x1,y1), (x2,y2))"
BOX	STRING (255) "((x1,y1), (x2,y2))"
PATH	CLOB "((x1,y1), (xn,yn))"
POLYGON	CLOB "((x1,y1), (xn,yn))"
CIRCLE	STRING (255) "(x,y),r"
JSON	NCLOB
<div style="border: 1px solid #ccc; padding: 10px; margin: 5px 0;">  <p><i>Reading and writing JSON data is performed using wide string (where each character occupies 2 bytes). Therefore, when replicating JSON columns to a PostgreSQL target, set the limited lob size to twice the maximum JSON length. So, for example, if the JSON data is 4 KB, set the limited lob size value to 8 KB.</i></p> </div>	
JSONB	NCLOB
<div style="border: 1px solid #ccc; padding: 10px; margin: 5px 0;">  <p><i>Reading and writing JSONB data is performed using wide string (where each character occupies 2 bytes). Therefore, when replicating JSONB columns to a PostgreSQL target, set the limited lob size to twice the maximum JSONB length. So, for example, if the JSONB data is 4 KB, set the limited lob size value to 8 KB.</i></p> </div>	
ARRAY	NCLOB
COMPOSITE	NCLOB

Google Cloud SQL for PostgreSQL data types	Qlik Replicate data types
INT4RANGE	STRING (255)
INT8RANGE	STRING (255)
NUMRANGE	STRING (255)
TSRANGE	STRING (255)
INT4MULTIRANGE	STRING (255)
INT8MULTIRANGE	STRING (255)
NUMMULTIRANGE	STRING (255)
TSMULTIRANGE	STRING (255)
CHARACTER VARYING	If length is specified: WSTRING (LENGTH) If no length is specified: WSTRING (8000)
TINTERVAL	WSTRING(255)



The PostGIS data type is not supported.

Homogeneous Replication

The following section describes how Replicate handles replication from Google Cloud SQL for PostgreSQL to the following targets: PostgreSQL, Microsoft Azure Database for PostgreSQL, Google Cloud SQL for PostgreSQL, and Google Cloud AlloyDB for PostgreSQL.

In homogeneous replication, source column and table collations will be replicated to the target as described in *Column and table collation* (page 242).



In homogeneous replication, the source data first passes through the Qlik Replicate data type and is therefore subject to any limitations of that type.

For information on Replicate data types and their limitations (where relevant), see Replicate data types (page 81).

For information on which Replicate data types the source data passes through when replicating from Google Cloud SQL for PostgreSQL, see the Google Cloud SQL for PostgreSQL to Qlik Replicate data types mapping table described earlier.

Limitations and considerations


Homogeneous replication is not supported with the following Replicate configurations or operations:

- Apply Changes only tasks
- Stopping and resuming a task
- Starting a task from a timestamp
- When the **if target table already exists** in the task settings' **Full Load Settings** tab is set to **Do nothing**

Data type considerations and exceptions

When replicating from Google Cloud SQL for PostgreSQL to a PostgreSQL-based target, source and target data types will be identical, with the following exceptions:

Data type exceptions in homogeneous replication

Google Cloud SQL for PostgreSQL source	PostgreSQL-based target
ENUM	STRING
COMPOSITE	STRING
NUMERIC	NUMERIC (28,6)
NUMERIC (p,s) A numeric array with precision and scale.	NUMERIC A numeric array without precision and scale.
TINTERVAL	 <p>As PostgreSQL have discontinued support for the TINTERVAL data type, tables with this data type will not be created on the target if the target database version does not support it.</p>
INT4MULTIRANGE	STRING (255)
INT8MULTIRANGE	STRING (255)
NUMMULTIRANGE	STRING (255)
TSMULTIRANGE	STRING (255)

See also the note about JSON data type replication in *Supported data types* (page 237).

Column and table collation

When replicating from Google Cloud SQL for PostgreSQL to a PostgreSQL-based target, column and table collations will be replicated to the target.



To support collation replication, the DBA must ensure that the collations defined for the Google Cloud SQL for PostgreSQL database are the same as those defined for the PostgreSQL-based target database.

Non-nullable columns and Primary/Unique index names

Non-nullable columns and Primary/Unique Index names are preserved during homogeneous replication.

Replication of Partitioned Tables

In order to replicate partitioned tables from a Google Cloud SQL for PostgreSQL source to a PostgreSQL-based target, you need to do the following:

1. Manually create the parent and child tables in the target database.
2. Define a separate task to replicate to those tables.
3. Set the **Truncate before loading** option in the *Full Load Settings* (page 1171).

Setting general connection properties

This section describes how to configure general connection properties.

To add a Google Cloud SQL for PostgreSQL endpoint source database to Qlik Replicate:

1. In the Qlik Replicate console, click **Manage Endpoints Connections** to open the **Manage Endpoints Connections** dialog box. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints* (page 114).
2. In the **Name** field, type a name for your Google Cloud SQL for PostgreSQL database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Google Cloud SQL for PostgreSQL database. This is optional.
4. Select **Source** as the database **role**.
5. Select **Google Cloud SQL for PostgreSQL** as the database **Type**.
6. Type the **Server** name. This is the name or IP address of the computer with the Google Cloud SQL for PostgreSQL database that you want to access.
7. Optionally, change the default port (5432).
8. Enter the Google Cloud SQL for PostgreSQL database authentication information (**User name**, **Password**) of an authorized Google Cloud SQL for PostgreSQL user.



Consider the following:

- This information is case sensitive.
- To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**. If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box. To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

9. Type the **Database** name or browse for a database.

SSL authentication options

In the **Security** section, you can configure Replicate to connect to the PostgreSQL database using SSL.

- **SSL Mode:** Select one of the following:
 - **Disable** - Connect with a surname and password only.
 - **Allow** - Establish an encrypted connection if requested by the server.
 - **Prefer** - Establishes an encrypted connection if the server supports encrypted connections, falling back to an unencrypted connection if an encrypted connection cannot be established.
 - **Require** - Establishes an encrypted connection if the server supports encrypted connections. The connection attempt fails if an encrypted connection cannot be established.
 - **Verify CA** - Similar to **Required**, but also verifies the server Certificate Authority (CA) certificate against the configured CA certificates. The connection attempt fails if no valid matching CA certificates are found.
 - **Verify Full** - Similar to **Verify CA**, but also performs host name identity verification by checking the host name the client (i.e. Replicate) uses for connecting to the server against the identity in the certificate that the server sends to the client. The client checks whether the host name that it uses for connecting matches the Common Name value in the server certificate. The connection fails if there is a mismatch.
- **Client certificate path** - The path to the client certificate requested by the server.
- **Secret key path** - The path to the client private key file in PEM format.
- **CA path** - The path to the Certificate Authority (CA) that issued the client certificate file in PEM format.
- **CRL path** - The path to the CRL certificate. This file contains certificates revoked by certificate authorities. If the server certificate appears in this list, the connection will fail.
- **SSL compression** - Select this option to compress the data before it is encrypted.

Setting advanced connection properties

In the **Advanced** tab, you can configure CDC parameters, set internal parameters, and view a summary of your settings.

- **Capture DDL events:** When this option is selected, the following actions occur:
 - Operational artifacts are created (by Replicate) in the database when the task starts. In order to capture DDL events, Qlik Replicate creates various artifacts in the Google Cloud SQL for PostgreSQL database when the task starts. You can later remove these artifacts as described in *Removing Replicate artifacts from the Google Cloud SQL for PostgreSQL database (page 246)*.
 - Streamed DDL events are captured.
- **Create DDL artifacts in schema:** The schema in which the operational DDL database artifacts will be created. The default value is "Public".
- **WAL heartbeat:** An Apply Changes task that is running but not capturing changes (due to source table inactivity) will continue to occupy the LSN position in its replication slot, thereby preventing truncation of the WAL. Since the WAL is a server-wide resource used by all Google Cloud SQL for PostgreSQL processes, it may grow extremely large if no changes are captured for an extended period. To prevent this from happening, enable the "WAL heartbeat" option. When this option is enabled, the Google Cloud SQL for PostgreSQL source endpoint mimics task activity by periodically committing pseudo transactions (i.e. "Heartbeats") to the heartbeat table, thereby advancing the task slot's LSN position.
 - **Schema for heartbeat artifacts:** The schema in which the WAL heartbeat table (`attrep_wal_heartbeat`) will be created. The default value is "public".
 - **Heartbeat frequency (minutes):** The frequency with which to commit transactions to the heartbeat table.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

Removing Replicate artifacts from the Google Cloud SQL for PostgreSQL database

In order to capture DDLs, Qlik Replicate creates various artifacts in the Google Cloud SQL for PostgreSQL database when the task starts. When the task completes, you may wish to remove these artifacts.

To remove the artifacts, issue the following statements (in the order they appear below), where `public` is the default schema in which the artifacts were created:

- `drop event trigger attrep_intercept_ddl`



The event trigger does not belong to a specific schema.

- `drop function public.attrep_intercept_ddl()`
- `drop table public.attrep_ddl_audit`
- `drop schema public`
- `drop table schema.attrep_wal_heartbeat`

where `schema` is the **Schema for heartbeat artifacts** specified in the **Advanced** tab of the endpoint settings.



Dropping a schema should be done with extreme caution, if at all. Never drop an operational schema, especially not `public`.

9.10 Using Hadoop as a source

This section describes how to set up and use Hadoop as the source endpoint in a replication task.

In this section:

- *Prerequisites (page 248)*
- *Limitations and considerations (page 248)*
- *Required permissions (page 248)*
- *Hadoop Endpoint source data types (page 248)*
- *Setting general connection properties (page 249)*
- *Setting advanced connection properties (page 253)*

Prerequisites

Before you begin to work with a Hadoop cluster as a data source in Qlik Replicate, make sure that the following prerequisites have been met:

- **General:**
 - The Hadoop WebHDFS must be accessible from the Qlik Replicate machine.
 - The Hadoop Data Nodes must be accessible from the Qlik Replicate machine.
 - The Hadoop WebHDFS service must be running.
 - To access Hive using WebHCat, the Hadoop WebHCat service must be running. Other methods for accessing Hive are described later in this chapter.
 - The user specified in the Qlik Replicate Hadoop target settings must have access to HiveServer2.
- **SSL:** Before you can use SSL, you first need to perform the following tasks:
 - Configure each NameNode and each DataNode with an SSL certificate (issued by the same CA).
 - Place the CA certificate on the Replicate Server machine. The certificate should be a base64-encoded PEM (OpenSSL) file.
- **Permissions:** The user specified in the Hadoop source settings must have read permission for the HDFS directories that contain the data files.

Limitations and considerations

The following limitations apply:

- Change data capture is not supported. The Hadoop endpoint will not be available for selection if either **Apply Changes** or **Store Changes** is enabled in the task settings.
- Replicating data compressed using a method other than gzip is currently not supported.
- Replicating data that is not in Text format is currently not supported.
- Replicating from tables with skews, buckets or partitions is currently not supported.
- Limited LOB support only.
- Replicating data with 4-byte emoji characters is not supported.

Required permissions

The Hadoop NameNode must be accessible from the Qlik Replicate machine.

Hadoop Endpoint source data types

The following table shows the Hadoop data types that are supported when using Qlik Replicate and the default mapping to Qlik Replicate data types.

For information on how to view the data type that is mapped in the target, see the section for the target endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).

Hadoop and Qlik Replicate data types

Hadoop data types	Qlik Replicatedata types
BOOLEAN	BOOL
BINARY	BLOB
DATE	DATE
TIMESTAMP	DATETIME
TINYINT	INT1
SMALLINT	INT2
INT	INT4
BIGINT	INT8
FLOAT	REAL4
DOUBLE	REAL8
VARCHAR	STRING
CHAR	STRING
STRING	CLOB
DECIMAL	NUMERIC

Unsupported data types

The Complex Types listed below are not supported:

- ARRAYS
- MAPS
- STRUCTS
- UNION

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a Hadoop source endpoint to Qlik Replicate:

1. In the Qlik Replicate console, click the Manage Endpoint Connections toolbar button to open the Manage Endpoints Connections dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.
2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the endpoint being used.
3. In the **Description** field, type a description that helps to identify the Hadoop endpoint. This is optional.
4. Select **Source** as the endpoint **Role**.

5. Select **Hadoop** as the endpoint **Type**.
6. In the **Hadoop NameNode** field, enter the host name or IP address of the Hadoop NameNode machine.



Consider the following:

- *This information is case sensitive.*
- *To determine if you are connected to the endpoint you want to use or if the connection information you entered is correct, click **Test Connection**. If the connection is successful, a green confirmation message is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box. To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.*

7. In the **Security section**, do the following:
 - a. To encrypt the data between the Replicate machine and HDFS, select **Use SSL**. In order to use SSL, first make sure that the [SSL](#) prerequisites described in *Prerequisites (page 248)* has been met.
In the **CA path** field, either specify the directory containing the CA certificate.
-OR-
Specify the full path to a specific CA certificate.
 - b. Select one of the following authentication types:
 - **User name** - Select to connect to the Hadoop cluster with only a user name. Then, in the **User name** field, specify the name of a user authorized to access the Hadoop cluster.
 - **Kerberos** - Select to authenticate against the Hadoop cluster using Kerberos. Replicate automatically detects whether Qlik Replicate Server is running on Linux or on Windows and displays the appropriate settings.
Qlik Replicate Server on Linux:
When Qlik Replicate Server is running on Linux, select either **Ticket** or [Keytab](#) from the **Kerberos options** drop-down list.
If you selected **Ticket**, select one of the following options:
 - **Use global Kerberos ticket file** - Select this option if you want to use the same ticket for several Hadoop endpoints (source or target). In this case, you must make sure to select this option for each Hadoop endpoint instance that you define.
 - **Use specific Kerberos ticket file** - Select this option if you want to use a different ticket file for each Hadoop endpoint. Then specify the ticket file name in the designated field.
This option is especially useful if you need to perform a task-level audit of Replicate activity (using a third-party tool) on the Hadoop NameNode. To set this up, define several instances of the same Hadoop endpoint and specify a unique Kerberos ticket file for each instance. Then, for each task, simply select a different Hadoop endpoint instance.



- You need to define a global Kerberos ticket file even if you select the **Use specific Kerberos ticket file** option. The global Kerberos ticket file is used for authentication when selecting a Hive endpoint, when testing the connection (using the **Test Connection** button), and when selecting which tables to replicate.
- When replicating from a Hadoop source endpoint to a Hadoop target endpoint, both endpoints must be configured to use the same ticket file.

For additional steps required to complete setup for Kerberos ticket-based authentication, see *Using Kerberos authentication* (page 792).

If you selected **Keytab**, provide the following information:

- **Realm:** The name of the realm in which your Hadoop cluster resides. For example, if the full principal name is `john.doe@EXAMPLE.COM`, then `EXAMPLE.COM` is the realm.
- **Principal:** The user name to use for authentication. The principal must be a member of the realm entered above. For example, if the full principal name is `john.doe@EXAMPLE.COM`, then `john.doe` is the principal.
- **Keytab file:** The full path of the Keytab file. The Keytab file should contain the key of the **Principal** specified above.

Qlik Replicate Server on Windows:

When Qlik Replicate Server is running on Windows, select one of the following:

- **Use the following KDC:** Select **Active Directory** (default) if your KDC is Microsoft Active Directory or select **MIT** if your KDC is MIT KDC running on Linux/UNIX.



When the Replicate KDC and the Hadoop KDC are in different domains, a relationship of trust must exist between the two domains.

- **Realm:** The name of the realm/domain in which your Hadoop cluster resides (where *realm* is the MIT term while *domain* is the Active Directory term).
- **Principal:** The username to use for authentication. The principal must be a member of the realm/domain entered above.
- When **Active Directory** is selected - **Password:** The password for the principal entered above.
- When **MIT** is selected - **Keytab file:** The keytab file containing the principal entered above.



When replicating from a Hadoop source endpoint to a Hadoop target endpoint, both endpoints must be configured to use the same parameters (KDC, realm, principal, and password).

9 Adding and managing source endpoints

If you are unsure about any of the above, consult your IT/security administrator. For additional steps required to complete setup for Kerberos authentication, see *Using Kerberos authentication (page 792)*.

- **User name and password** - Select to connect to the Hadoop NameNode or to the Knox Gateway (when enabled - see below) with a user name and password. Then, in the **User name** and **Password** fields, specify the required user name and password.



Consider the following:

- A user name and password is required to access the MapR Control System.
- This information is case sensitive.



*Make sure that the specified user has the required Hadoop access privileges. For information on how to provide the required privileges, see *Required permissions (page 248)*.*

8. If you need to access the Hortonworks Hadoop distribution through a Knox Gateway, select **Use Knox Gateway**.



*To be able to select this option, first select **Use SSL** and then select **Password** from the **Authentication type** drop-down list.*

9. Provide values for the following fields:

- **Knox Gateway host** - The FQDN (Fully Qualified Domain Name) of the Knox Gateway host.
- **Knox port** - The port number to use to access the host. The default is "8443".
- **Knox Gateway path** - The context path for the gateway. The default is "gateway".



*The port and path values are set in the **gateway-site.xml** file. If you are unsure whether the default values have been changed, contact your IT department.*

- **Cluster name** - The cluster name as configured in Knox. The default is "default".

10. In the **HDFS section**, select either **WebHDFS** or **HttpFS** as the HDFS access method. If you are accessing MapR, it is recommended to use HttpFS.



*When the **Use Knox Gateway** option is selected, the NameNode, **HttpFS Host**, and **Port** fields described below are not relevant (and are therefore hidden).*

11. Do one of the following, depending on whether you selected **WebHDFS** or **HttpFS**:
If you selected **WebHDFS**:

1. In the **NameNode** field, specify the IP address of the NameNode.



This is the Active node when High Availability is enabled (see below).

2. Replicate supports replication from an HDFS High Availability cluster. In such a configuration, Replicate communicates with the Active node, but switches to the Standby node in the event of failover. To enable this feature, select the **High Availability** check box. Then, specify the FQDN (Fully Qualified Domain Name) of the Standby NameNode in the **Standby NameNode** field.
3. In the **Port** field, optionally change the default port (50070).

If you selected **HttpFS**:

1. In the **HttpFS Host** field, specify the IP address of the HttpFS host.
2. In the **Port** field, optionally change the default port (14000).

12. In the **Hive Access section**, do the following:



*When the **Use Knox Gateway** option is selected, the **Host** and **Port** fields described below are not relevant (and are therefore hidden).*

- **Access Hive using** field (WebHCat): This value cannot be changed.
- **Host** field: Specify the IP address of the Hive machine.
- **Port** field: Optionally change the default port.
- **Database** field: Specify the name of the Hive target database.

Setting advanced connection properties

In the **Advanced** tab, you can specify source file delimiters and other properties. Note that the source file delimiters only need to be specified if the SerDe property names used to create the source data files are different from the Hadoop defaults.

The default Hadoop property names are as follows:

- field.delim (**Field delimiter** in the UI)
- serialization.null.format (**Null value** in the UI)
- escape.delim (**Escape character** in the UI)
- line.delim (**Record delimiter** in the UI)
- quote.delim (**Quote character** in the UI)

In the **Advanced** tab, you can set the parameters described in the following table.

Advanced tab options

Option	Description
Field delimiter	The delimiter used to separate fields in the source files.

Option	Description
Null value	<p>The value used to indicate a null value in the source files.</p> <p>Example (where @@ is the null value):</p> <pre>mike,male,295678 sara,female,@@</pre>
Escape character	<p>This can either be the character used to escape the field delimiter character - if the source files were created using a SerDe that does not support quote characters (see Example 1) or the character used to escape the quote character - if the source files were created using a SerDe that supports quote characters (see Example 2).</p> <p>Example 1 (where \ is the escape character and a comma is the field delimiter):</p> <pre>sunroof\,power-steering</pre> <p>Example 2 (where \ is the escape character and double quotes is the quote character):</p> <pre>"\"sunroof, power-steering\""</pre>
Record delimiter	<p>The delimiter used to separate records (rows) in the source files.</p> <p>If the LazySimpleSerde SerDe was used, the record delimiter must be \n.</p>
Quote character	<p>The character used to escape the field delimiter character in the source files. When a field delimiter is escaped, it is interpreted as actual data, and not as a field delimiter.</p> <p>Example (where double-quotes is the quote character):</p> <pre>"sunroof,power-steering"</pre>

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

9.11 Using IBM DB2 for iSeries as a source

This section describes how to set up and use an IBM DB2 for iSeries database as the source database in a replication task.

In this section:

- *Prerequisites (page 257)*
- *Required permissions (page 259)*
- *Limitations and considerations (page 259)*
- *Supported data types (page 260)*
- *Setting general connection properties (page 261)*
- *Setting advanced connection properties (page 262)*

Prerequisites

The following section describes the prerequisites for working with the Qlik Replicate IBM DB2 for iSeries endpoint.



iAccess ODBC driver 7.1.0. is not supported.

Client

The following section explains how to install the driver required by the IBM DB2 for iSeries endpoint.

Replicate on Windows

1. Download the **IBM i Access Client Solutions Windows Application Package** from the IBM website. For more information, see <https://www.ibm.com/support/pages/ibm-i-access-client-solutions-5733xj1>
2. Install the downloaded package on the Replicate server.
This will install the **iSeries Access ODBC Driver** that is used by Replicate. The version supported by Replicate is 13.64.xx.
After the installation completes, an entry will be added to the **Drivers** tab in the Windows **ODBC Data Source Administrator** application.

Example:

IBM i Access ODBC Driver version 13.64.26



The displayed driver version will differ according to where you view it. For instance, assuming the driver version is as in the above example, then:

- *In the Replicate task logs, the driver will appear as **ODBC driver version: '07.01.026'**.*
- *In the package's readme file, the version will appear as **5733-XJ1 Version: 1.1.0.26***

Replicate on Linux

1. Download the **IBM i Access Client Solutions Linux Application Package** from the IBM website. For more information, see <https://www.ibm.com/docs/en/i/7.2?topic=packages-linux-application-package>
2. Stop the Replicate service:
`./areplicate stop`
3. Optionally, confirm that the service has stopped:
`./areplicate status`
4. Install the RPM that is included in the downloaded package.

Example:

```
rpm -i ibm-iaccess-1.1.0.26-1.0.x86_64
```



*The displayed driver version will be different in the Replicate task logs. For instance, assuming the driver version is as in the above example, it will appear as **ODBC driver version: '07.01.026'** in the Replicate task logs.*

5. Change the working directory to `<Replicate_Install_Dir>/bin`.
6. Add the DB2 driver location to the Linux library path.
To do this, add the following line to the **site_arep_login.sh** file in `<install_dir>/replicate/bin`:

```
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/path/lib64
```

where `path` is the path to the driver.
Example:

```
echo "export LD_LIBRARY_PATH=\$LD_LIBRARY_PATH:/opt/ibm/iaccess/lib64" >> site_arep_login.sh
```
7. Optionally, confirm that the driver location was copied:

```
cat site_arep_login.sh
```
8. After the installation completes, open the **/etc/odbcinst.ini** file and make sure it contains the following section:

```
[IBM i Access ODBC Driver 64-bit]
Description = IBM i Access for Linux 64-bit ODBC Driver
Driver = /opt/ibm/iaccess/lib64/libcwbodbc.so
Setup = /opt/ibm/iaccess/lib64/libcwbodbc.so
Threading = 0
DontDLClose = 1
UsageCount = 1
```
9. Start the Replicate service:

```
./areplicate start
```
10. Optionally, confirm that the service has started:

```
./areplicate status
```

Change processing

Journal prerequisites

All of the source tables for a given replication task need to be journaled to the same journal. The name of the journal and the library in which it is located must be specified in the endpoint settings. During the task, Replicate polls this journal for changes to the source tables.

When you start journaling the source tables, the `Record_images` parameter can be set to `*BOTH` (for capturing before and after images) or `*AFTER`.

When the `Record_images` parameter set to `*AFTER`, deleted records will contain NULL values in their data fields. To save a copy of the deleted record to the deleted record journal, you must set the `Record_images` parameter set to `*BOTH`.



If you need to run several replication tasks (that replicate data from IBM DB2 for iSeries), it is more efficient (though not essential) to create a separate journal for each task. As only one journal can be specified per endpoint, you would also need to define a separate endpoint for each task.

MINENTDTA parameter

The `MINENTDTA` parameter must be set to `*NONE` as `*FILE` (minimal logging) is not supported.

Required permissions

The following permissions must be granted to the user specified in the **General** tab of the IBM DB2 for iSeries endpoint settings:

- USER CLASS = *USER (default value)
- Special authority = *NONE
- Full Load: Read permissions for the source tables.
- CDC: Read permissions for the journal defined for the IBM DB2 for iSeries endpoint and for the task's source tables.
- SELECT permission on the following tables:
 - QSYS2.SYSCOLUMNS
 - QSYS2.SYSTABLES

You must also set the following **Authorities and Locks** for the IBM DB2 for iSeries database:

- **Journal Authority:** *USE
- **Journal Library Authority:** *EXECUTE
- **Journal Receivers Authority:** *USE
- **Journal Receivers Library's Authority:** *EXECUTE
- **File Authority (if specified):** *USE
- **File Library Authority:** *EXECUTE
- **Journal Lock:** *SHRRD
- **Journal Receiver Lock:** *SHRRD
- **File Lock (if specified):** *SHRRD

*OBJEXIST is also required for the journal authority if any of the following are true:

- *ALLFILE has been specified for the file key.
- Specified object does not exist on the system.
- *IGNFILSLT or *IGNOBSLT is specified for the journal code selection value for any selected journal codes.

DB2 server configuration

In rare situations, the following error might be encountered when using the IBM DB2 for iSeries endpoint in a replication task:

```
SQL0666N SQL query exceeds specified time limit or storage limit
```

To prevent this error from occurring, set the following attributes for the Replicate Job/UserID(s):

```
QRYSTGLMT=*NOMAX
```

```
QRYTIMLMT=*NOMAX
```

For more information, see https://www.ibm.com/docs/en/i/7.4?topic=ssw_ibm_i_74/cl/chgqrya.htm

Limitations and considerations

You should be aware of the following limitations and considerations when using an IBM DB2 for iSeries source.

- Multi-member (Partitioned) tables are not supported
- Field level encryption is not supported
- Journal receivers must be System Managed rather than User Managed.
- Journals containing minimized data are not supported.
The MINENTDTA parameter should be set to: MINENTDTA(*NONE)
- All tables must be in the same journal
- The XML data type is not supported
- The DBCLOB data type is not supported
- Row size cannot exceed 32740 bytes
- Usually, Replicate only applies source database changes to the target if the transactions with the changes were committed before the Full Load started. However, records that were deleted from the source as part of an uncommitted transaction will not be replicated to the target during Full Load.
- DELETE operations in auto-commit mode will be ignored for tables journaled with *AFTER images. In such cases, DELETE operations will only be captured if one of the [Add RRN column](#) options is enabled and is set as the only Primary Key.
- When auto-commit=ON, the timestamp of the last committed transactions will be displayed as "01-01-1970 00:00" in the SOURCE_CURRENT_TIMESTAMP column of the [attrep_status Control](#) table.
- Replicating columns, tables, and libraries with special characters in their name is not supported.
- **Tables created by DDS:** If you need to replicate tables created by DDS, please seek guidance from Qlik Support or your Qlik Customer Success Manager.
- Replicating data with 4-byte emoji characters is not supported.
- When the **Use table and schema system names** option is enabled (in the endpoint's **Advanced** tab), only table/schema names that contain uppercase letters, digits, and underscores (_) are supported. For more information on this option, see *Replicating System Names (page 265)*.
- Remote journals are not supported.
- The DROP TABLE DDL is not supported
- The RENAME TABLE DDL is not supported

Supported data types

The following table shows the IBM DB2 for iSeries database source data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.

For information on how to view the data type that is mapped in the target, see the section for the target database you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

Supported data types

IBM DB2 for iSeries Source Data Types	Qlik Replicate Data Types
INTEGER	INT4
SMALLINT	INT2

IBM DB2 for iSeries Source Data Types	Qlik Replicate Data Types
BIGINT	INT8
DECIMAL (p,s)	NUMERIC (p,s)
FLOAT	REAL8
DOUBLE	REAL8
REAL	REAL4
CHAR (n)	If n<=32 KB, then: STRING
VARCHAR (n)	If n<=32 KB, then: STRING
GRAPHIC (n)	If n<=16 KB, then: STRING
VARGRAPHIC (n)	If n<=16 KB double byte chars, then: STRING
DATE	DATE
TIME	TIME
TIMESTAMP	DATETIME (6)
BLOB	BLOB Maximum size: 2 GB
CLOB	CLOB Maximum size: 2 GB
ROWID	BYTES - This should be a user-defined column.
DATALINK	STRING
TIMESTAMP WITH TIME ZONE	NOT SUPPORTED

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.



You can also use IBM DB2 for iSeries File Channel files as a source. For more information, see Using the Qlik Replicate file channel (page 1061).

To add an IBM DB2 for iSeries source endpoint to Qlik Replicate:

1. In the Qlik Replicate console, click **Manage Endpoint Connections** to open the **Manage Endpoint Connections** dialog box.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the IBM DB2 for iSeries database. This is optional.
4. Select **Source** as the database **role**.
5. Select **IBM DB2 for iSeries** as the database **Type**.



*When this endpoint is used as a duplicated source in a Log Stream Staging setup, select the **Read changes from log stream** check box and then select the relevant Log Stream Staging task from the drop-down list.
For information on setting up and managing Log Stream Staging tasks, see *Using the Log Stream* (page 1072).*

6. Choose one of the following:
 - **Use database alias (DSN)** - If you choose this option, specify the IBM DB2 for iSeries database ODBC DSN.
 - **Use these connection properties** - If you choose this option, enter the IBM DB2 for iSeries **Server** (hostname or IP address).
7. Enter the IBM DB2 for iSeries authentication information (**User Name, Password**) for the authorized user for this IBM DB2 for iSeries database. If you do not know this information, see your IBM DB2 for iSeries database administrator (DBA).



This information is case sensitive.



Make sure that the specified user has the required access privileges.

8. In the **Journal Name** field, enter the name of the journal containing the source tables.
See also: [Change Processing](#) prerequisites.
9. In the **Journal Library** field, enter the name of the library where the journal is located.
See also: [Change Processing](#) prerequisites.

Setting advanced connection properties

In the **Advanced** tab, you can set advanced connection properties such as overriding CCSID to Character Set mapping and setting internal Replicate parameters

Using the R4I UDTF for Change Capture

By default, changes captured from IBM DB2 for iSeries are filtered on the Replicate machine using the standard display_journal function. When there is a high volume of changes, this may impact Change Processing performance.


9 Adding and managing source endpoints

Installing the R4I UDTF on IBM DB2 for iSeries enables captured changes to be filtered on IBM DB2 for iSeries instead of on the Replicate machine, significantly improving CDC performance.



The R4I UDTF can capture up to 300 tables. If you need to capture more than 300 tables, do not use this option.

To set up change capture using the R4I UDTF:

1. Go to  [Product Downloads](#).
2. Select **Qlik Data Integration**.
3. Scroll down the **Product** list and select Replicate.
4. In the **Download Link** column, locate the **QlikReplicate_<version-build>_R4I.zip** file. Before starting the download, check the **Version** column to make sure that the R4I version you are downloading corresponds to your current Replicate version.
5. Install the R4I UDTF on the IBM DB2 for iSeries server.
6. Select the **Enable UDTF capturing** option.
7. If you changed the default CDC reader UDTF name during the R4I installation, specify the new name in the **CDC reader UDTF name** field.
8. In the **UDTF result set size (MB)** field, you can set the maximum size of the result-set buffer returned by the R4I UDTF. A larger result set will usually improve performance, but will also impact DB2 memory consumption. Therefore, if you want to increase the result-set size, best practice is to run the task in a test environment first as insufficient memory on DB2 might result in data inconsistency on the target (due to partial records being captured).

Checking for changes

Check for changes every (sec): How often to check the source database for changes.

Overriding CCSID to character set mapping

In some cases, character data in source tables may be encoded in a different CCSID than what is declared in the source database table definition. For example, a specific table or column definition might indicate that it uses CCSID 500 (EBCDIC International) whereas in fact, it uses CCSID 1148 (ENCODIC International with EURO). In this case, you can tell Replicate that the source definition CCSID 500 should be treated as CCSID 1148 (specifically, the character set named IBM-1148).

Note that when the source table definition specifies CCSID 65535 (meaning character set is unknown), you must specify what character set should be assumed when reading data from that table or column.



If there is a conflict between the character set mapping for a specific column and the character set mapping defined in the endpoint settings, the column-level character set mapping takes precedence.

For more information on overriding character set mapping at column level, see [Using the Transform tab](#) (page 1086).

To do this:

1. In the **Override CCSID to Character Set Mapping** section, click the **New** button.
A row is added to the table.
2. Enter the CCSID in the **CCSID** column and the code page in the **Character set** column.
The **Character set** (code page) must be specified in the proper format (e.g. ibm-1148_P100-1997). For a list of valid formats, see [the ICU Converter Explorer](#).
3. Repeat to map additional CCSID values.

Converting to a custom code page

Perform the following procedure if your source endpoint tables are defined with an incorrect CCSID and the correct definition is actually in a UCM file.

1. Create a mapping data file with the file extension **.ucm**.



If you edit an existing UCM file, you must also change the values of the `<code_set_name>` and `<icu:alias>` properties. If the file does not contain an `<icu:alias>` property, then you only need to change the value of the `<code_set_name>` property.

2. Create a CNV file for the UCM file by running the following command:

```
<product_dir>\bin\makeconv.exe -v <file_name>.ucm
```

Example:

```
"c:\Program Files\Attunity\Replicate\bin\makeconv.exe" -v 1047_EX.ucm
```

This will create a CNV file with the same name as the UCM file (for example, `1047_EX.cnv`).

3. Create a new subfolder named **icudt<XX>l** under `<product_dir>\bin` where the **XX** is the same as the number in the **icudt<XX>.dll** file name.
For example, If the DLL file name is **icudt69.dll**, create a new subfolder named **icudt69l**.
4. Copy the CNV file to the subfolder you just created (`<product_dir>\bin\icudt69l` in the example above).



When using the Replicate File Channel, the file should be copied to the same location on both Replicate servers.

5. Add a new character set mapping as follows:
 - a. In **CCSID** column, enter the original source CCSID number (e.g. **1047**)
 - b. In the **Character set column**, enter the name of the CNV file without the extension (e.g. **1047_EX**).
6. Restart the Qlik Replicate UI Server service.

Adding the RRN Column to Target Tables

Source tables that do not have a primary key, a unique index, or a combination of columns that can be used as a unique index, must be registered using the relative record numbers (RRN).

Select one the following options:

- **Add RRN column to target tables without a primary key or unique index**
- **Add RRN column to all target tables**

- **Do not add RNN column to target tables**

To support DELETE operations in auto-commit mode for tables journaled with *AFTER images, do the following:

1. Enable the **Add RNN column to all target tables** option.
2. Mark the new RNN column on the target as the table's sole Primary Key (using a transformation). Note that adding an RNN column without marking it as a Primary Key will result in the table being suspended if a DELETE operation is performed.

When you select one of the "Add RNN columns" options, both the Change Tables and the target tables will have an extra column, ATTREP_RRN of type INTEGER, which contains a unique value for each row. This column contains the RRN that corresponds to each source table row.



The IBM DB2 for iSeries RRN values represent the exact position of the row in the file structure that holds the table data. When a table is reorganized, the table is rebuilt resulting in new RRNs being allocated to each row. Consequently, reorganization of tables where the RRN is being used as a target key should be avoided whenever possible. If such reorganization is unavoidable, you should immediately reload the reorganized table to prevent unpredictable behavior when changes are applied to the target (e.g. failed DELETES, duplicate INSERTs, and so on).

Replicating System Names

The IBM DB2 for iSeries source endpoint replicates tables based on their SQL names (unlimited length). If your IBM DB2 for iSeries database does not use SQL names, it's likely that you'll want to keep the replication based on system names.

To do this, select the **Use table and schema system names** check box.

Skipping Journal Validation

From IBM DB2 for iSeries 7.3, Replicate automatically validates the specified journal. This involves checking that the journal exists and that it contains the tables selected for replication. When numerous tables are selected for replication, this process may take some time. In such cases, if you are sure that the specified journal exists and that it contains the correct tables, you may want to skip the validation phase.

To do this, select the **Skip journal validation** check box.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.

4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

9.12 Using IBM DB2 for LUW as a source

This section describes how to set up and use an IBM DB2 for LUW database as the source database in a replication task.

In this section:

- *Prerequisites (page 268)*
- *Replicating 4-byte UTF8 emojis (page 269)*
- *Limitations and considerations (page 269)*
- *Supported data types (page 270)*
- *Setting general connection properties (page 272)*
- *Setting advanced connection properties (page 273)*

Prerequisites


The following section describes the prerequisites for working with the IBM DB2 for LUW endpoint.

Driver setup

Supported drivers


Replicate supports driver versions 11.5.6 and 11.5.8 only. The setup instructions are for version 11.5.8.

Replicate on Windows

1. Download the IBM Data Server Client 11.5.8 (**v11.5.8_ntx64_client.exe**) from  [IBM Fix Central](#).
2. Install it on the Qlik Replicate Server machine.

Replicate server on Linux

The following steps need to be performed on the Replicate machine:

1. Download the IBM Data Server Driver for ODBC and CLI 11.5.8 (**v11.5.8_linuxx64_server_dec.tar.gz**) from  [IBM Fix Central](#).
2. Stop the Replicate service:

```
./areplicate stop
```
3. Optionally, confirm that the service has stopped:

```
./areplicate status
```
4. Install **v11.5.8_linuxx64_server_dec.tar.gz** and choose the “CLIENT” option.
5. If the Replicate machine does not have a DB2 instance, create a DB2 instance by running the following commands:

```
adduser <db2_instance_name>  
/opt/ibm/db2/v11.5/instance/db2icrt <db2_instance_name>
```
6. Change the working directory to *<Replicate_Install_Dir>/bin*.
7. Add the DB2 driver location to the Linux library path.
To do this, add the following line to the **site_arep_login.sh** file as follows:

```
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/path/lib
```

where path is the CLIENT path to the driver.
Example (with db2clnt1 as the instance name):

```
export LD_LIBRARY_PATH=$LD_LIBRARY_  
PATH:/home/db2clnt1/sqllib/lib:/home/db2clnt1/sqllib/lib64/gskit  
export DB2INSTANCE=db2clnt1
```
8. Optionally, confirm that the driver location was copied:

```
cat site_arep_login.sh
```
9. Add the following entry to the `odbcinst.ini` file under `/etc`:

```
[IBM DB2 ODBC DRIVER]  
Driver = /home/db2clnt1/sqllib/lib/libdb2o.so  
fileusage=1  
dontd1close=1
```
10. Start the Replicate service:

```
./areplicate start
```
11. Optionally, confirm that the service has started:

```
./areplicate status
```


12. Define the IBM DB2 for LUW connection settings as described in *Setting general connection properties* (page 272) and click **Test Connection**.

If you get the following error:

Cannot connect to DB2 LUW Server ODBC unknown error.

RetCode: SQL_ERROR SqlState: NativeError: -1390 Message: [unixODBC][IBM][CLI Driver] SQL10007N Message "0" could not be retrieved. Reason code: "3". ODBC general error.

- a. Run the following command:

```
/opt/ibm/db2/v11.5/bin /db2cli writecfg add -database <db_name_from_endpoint_connection_settings> -host <server_name_from_endpoint_connection_settings> -port <port_from_endpoint_connection_settings>
```
- b. Change to the Replicate**bin** directory (<install_dir>/bin). Then, stop and start the Replicate service:

```
./areplicate stop  
./areplicate start
```

IBM DB2 for LUW server prerequisites

To enable CDC (Change Data Capture):

- **Set the database to be recoverable** - To capture changes, Replicate requires the database to be configured as recoverable. A database is recoverable if either or both of the database configuration parameters LOGARCHMETH1 and LOGARCHMETH2 are not set to OFF.
- **Permissions** - The Qlik user must be granted the following permissions:
 - SYSADM or DBADM
 - DATAACCESS

- To enable data capture from IBM DB2 for LUW, the source tables need to be created as follows:

```
CREATE / ALTER TABLE table-name ... DATA CAPTURE CHANGES [INCLUDE LONGVAR COLUMNS];
```

You can either configure Replicate to perform this operation by selecting **Automatically enable Data Capture Changes (requires admin privilege)** in the endpoint settings or you can do this manually.



When the **Automatically enable Data Capture Changes (requires admin privilege)** option is selected, Replicate performs the `ALTER TABLE` operation when the task starts to run. If a new table that matches the table selection pattern is added (to the source database) while the task is already running, Replicate will only capture changes from it if it was created with the `DATA CAPTURE CHANGES` attribute described above.

Replicating 4-byte UTF8 emojis

In order to replicate 4-byte UTF8 emojis, you must transform the emojis from `STRING(n)` to `WSTRING(n)`.

For information on defining transformations, see *Using the Transform tab* (page 1086).

Limitations and considerations

When using IBM DB2 for LUW as a data source, the following limitations currently apply.

- Clustered database is not supported



Users can define a separate IBM DB2 for LUW database for each of the endpoints in the cluster.

- Changes to DECFLOAT columns will be ignored during CDC.
- The RENAME COLUMN statement is not supported.
- When performing updates on MDC (Multi-Dimensional Clustering) tables, each update is shown in the task monitor as INSERT + DELETE.
- **DB2 10.5 or later:** Variable-length string columns with data that is stored out-of-row will be ignored. Note that this limitation is only applicable to tables created with extended row size.
- The Purescale environment is supported using the **Start processing changes from source change position** Run option only.
- If you need to replicate 4-byte emoji characters, use a transformation to convert the data type from STRING(n) to WSTRING (n*2).
- When truncating a table with multiple partitions, the number of DDL events displayed in the Replicate console will be equal to the number of partitions. This is because IBM DB2 for LUW records a separate DDL for each partition.
- The following DDLs on partitioned tables are not supported:
 - ALTER TABLE ADD PARTITION
 - ALTER TABLE DETACH PARTITION
 - ALTER TABLE ATTACH PARTITION
- When the **Replicate LOB columns** task setting is disabled, any table that has LOB columns will be suspended during change processing.
- When the **Audit table** option is enabled in the **Store Changes Settings** tab, the first timestamp record in the audit table will be NULL.
- When the **Change table** option is enabled in the **Store Changes Settings** tab, the first timestamp record in the table may be Zero in some cases (i.e. 1970-01-01 00:00:00.000000).
- A DDL operation that occurs during Full Load, but that was not preceded by a DML operation will not be captured.

Supported data types

The following table shows the IBM DB2 for LUW database source data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.

For information on how to view the data type that is mapped in the target, see the section for the target database you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

9 Adding and managing source endpoints

Supported data types

IBM DB2 for LUW Source Data Types	Qlik Replicate Data Types
INTEGER	INT4
SMALLINT	INT2
BIGINT	INT8
DECIMAL (p,s)	NUMERIC (p,s)
FLOAT	REAL8
DOUBLE	REAL8
REAL	REAL4
DECFLOAT (p)	If precision = 16, then: REAL8 If precision is = 34, then: STRING
GRAPHIC (n)	If n<=127 KB, then: WSTRING
VARGRAPHIC (n)	If n<=16k double byte chars, then: WSTRING
LONG VARGRAPHIC	CLOB
CHAR (n)	STRING n<=255
VARCHAR (n)	STRING n<=32k
LONG VARCHAR (n)	CLOB n<=32k
CHAR (n) FOR BIT DATA	BYTES
VARCHAR (n) FOR BIT DATA	BYTES
LONG VARCHAR FOR BIT DATA	BYTES
DATE	DATE
TIME	TIME
TIMESTAMP	DATETIME
BLOB	BLOB

IBM DB2 for LUW Source Data Types	Qlik Replicate Data Types
CLOB	CLOB Maximum size: 2 GB
DBCLOB	CLOB Maximum size: 1 G double byte chars
XML	CLOB

Setting general connection properties

You can add an IBM DB2 for LUW database to Qlik Replicate to use as a source. For information on how to add endpoints, see *Defining and managing endpoints (page 114)*.



You can also use IBM DB2 for LUW File Channel files as a source. For more information, see *Using the Qlik Replicate file channel (page 1061)*.

To add an IBM DB2 for LUW source endpoint to Qlik Replicate:

1. In the Qlik Replicate console, click **Manage Endpoint Connections** to open the **Manage Endpoint Connections** dialog box.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the IBM DB2 for LUW database. This is optional.
4. Select **Source** as the database **role**.
5. Select **IBM DB2 for LUW** as the database **Type**.



When this endpoint is used as a duplicated source in a Log Stream Staging setup, select the **Read changes from log stream** check box and then select the relevant Log Stream Staging task from the drop-down list.

For information on setting up and managing Log Stream Staging tasks, see *Using the Log Stream (page 1072)*.

6. Choose one of the following:



When replicating from an IBM DB2 for LUW HADR environment and the task's **Apply Changes** replication option is enabled, the endpoint must be configured to read from the primary instance.

- **Use database alias (DSN)** - If you choose this option, specify the IBM DB2 for LUW database alias.

- **Use these connection properties** - If you choose this option, enter the IBM DB2 for LUW **Server** (hostname or IP address), **Port** and **Database name** in the designated fields.
7. Enter the IBM DB2 for LUW authentication information (**User Name**, **Password**) for the authorized user for this IBM DB2 for LUW database. If you do not know this information, see your IBM DB2 for LUW database administrator (DBA).



This information is case sensitive.



Make sure that the specified user has the required access privileges.

Setting advanced connection properties

In the **Advanced** tab, you can set the following properties:

- **Maximum Buffer Size for Read (KB)**: Specify the maximum number of bytes to read each time the log is accessed during Change Processing. If you encounter performance issues, adjusting this number may help.
- **Change Data Capture**: To enable data capture from IBM DB2 for LUW, the source tables need to be created as follows:

```
CREATE / ALTER TABLE table-name ... DATA CAPTURE CHANGES [INCLUDE LONGVAR COLUMNS];
```

You can either configure Replicate to perform this operation by selecting **Automatically enable DATA CAPTURE CHANGES (requires admin privilege)** in the endpoint settings or you can do this manually by selecting **Let DBA set up data capture**.
- **Check for changes every (sec)**: How often to check the source database for changes.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

Resuming or starting a task from LRI in a pureScale environment

To resume or start a task from LRI in a pureScale environment:

1. Run the db2pd DB2 command line to determine the Current Log Number:

```
$ db2pd -log -db database_name
```

where *database_name* is the name of your database.

The output will be similar to the following:

```
Database Member 0 -- Database SAMPLE -- Active -- Up 0 days 00:27:02 -- Date 2018-05-23-14.49.45.515600
```

Logs:

```
Current Log Number          2
Pages Written                19
Cur Commit Disk Log Reads  0
Cur Commit Total Log Reads 0
Method 1 Archive Status     n/a
Method 1 Next Log to Archive n/a
Method 1 First Failure      n/a
Method 2 Archive Status     n/a
Method 2 Next Log to Archive n/a
Method 2 First Failure      n/a
Log Chain ID                 0
Current LSO                   57142478
Current LSN                   0x000000000004FB14
```

Address	StartLSN	StartLSO	State Size	Pages	Filename
0x00007F164E99F090	00000000000429EC	48912001	0x00000000 1000	1000	S0000000.LOG
0x00007F16511319F0	000000000004901E	52988001	0x00000000 1000	1000	S0000001.LOG
0x00007F1651132350	000000000004F9A6	57064001	0x00000000 1000	1000	S0000002.LOG



The most recent (current) log is number 2.

2. Run the db2flsn command to determine the LRI/ LSN range of that log file:

```
$ db2flsn -db SAMPLE -lri range 2
```

The output will be similar to the following:

```
S0000002.LOG: has LRI range  
0000000000000001000000000000225400000000004F9A6 to  
000000000000000100000000000022cc00000000004FB13
```

In the above output, the task "Start From" LRI value would be:
0100000000000022cc00000000004FB13 (i.e. the 34 bytes at the end of the range).

Unable to retrieve the current LSN?

If you are unable to retrieve the current LSN, you can instruct Replicate to scan the log by specifying `currentLSN=scan` (as described in *Internal parameters (page 273)*) instead of the current LSN.



When `currentLSN=scan`, Replicate will search the log from the start until it finds the current LSN. This may take some time depending on the size of the log.

9.13 Using IBM DB2 for z/OS as a source

This section describes how to set up and use an IBM DB2 for z/OS database as the source endpoint in a replication task.

In this section:

- *Prerequisites (page 277)*
- *Limitations and considerations (page 280)*
- *Handling actions resulting in subtype 83 (page 281)*
- *Supported data types (page 282)*
- *Setting general connection properties (page 284)*
- *Setting advanced connection properties (page 285)*
- *Sample XMIT files “Receive” job (page 289)*
- *Installing and configuring R4Z (page 290)*


Prerequisites

The following section describes the prerequisites for working with the Qlik Replicate IBM DB2 for z/OS endpoint.

Install the R4Z product on z/OS

Before you can work with an IBM DB2 for z/OS source, you must first download, install and configure the R4Z component on z/OS. R4Z is available for download from Qlik Community.

To download R4Z:

1. Go to  [Product Downloads](#).
2. Select **Qlik Data Integration**.
3. Scroll down the **Product** list and select Replicate.
4. In the **Download Link** column, locate the **QlikReplicate_<version>_r4z** file. Before starting the download, check the **Version** column to make sure that the version correlates with the Replicate version you have installed.
5. Download R4Z and install it on your z/OS data source.

For installation, and configuration instructions, see *Installing and configuring R4Z (page 290)* .



*The Replicate and R4Z versions you install must be part of the same release, which can either be an initial release or a patch/service release. This means that if you upgrade Replicate, you must also upgrade your R4Z installation (to the version that correlates with the Replicate release you are installing), unless it has the same build number as your currently installed R4Z. Note that despite being part of the same release, the Replicate and R4Z build numbers (reflected in their file names) might not be the same. For example, Replicate May 2022 SR2 includes the following files: **QlikReplicate_2022.5.0.499_X64.zip** (the Replicate installer) and **QlikReplicate_2022.5.0.290_r4z.zip** (the R4Z installer).*

Driver setup

This section describes the client side and server side ODBC prerequisites. When installing on Linux, you can choose between installing the full ODBC client package or installing just the data server client.




The "enableWLB" driver option is not supported. To force the connection to automatically switch to another LPAR in a parallel sysplex (i.e. "failover"), set "enableACR" to "true" and "enableWLB" to "false" in the driver configuration.

Supported drivers

Replicate supports driver versions 11.5.6 and 11.5.8 only. The setup instructions are for version 11.5.8.

Replicate on Windows

1. Download the IBM Data Server Client 11.5.8 (**v11.5.8_ntx64_client.exe**) from  [IBM Fix Central](#).
2. Install it on the Qlik Replicate Server machine.

Replicate on Linux

Linux ODBC client package

If you prefer, you can install the data server client only as described in *Linux data server client* (page 278) below.

1. Download the IBM Data Server Driver for ODBC and CLI 11.5.8 (**v11.5.8_linuxx64_server_dec.tar.gz**) from  [IBM Fix Central](#).
2. Stop the Replicate service:

```
./areplicate stop
```
3. Optionally, confirm that the service has stopped:

```
./areplicate status
```
4. Install **v11.5.8_linuxx64_server_dec.tar.gz** and choose the “CLIENT” option.
5. If the Replicate machine does not have a DB2 instance, create a DB2 instance by running the following commands:

```
adduser <db2_instance_name>  
/opt/ibm/db2/v11.5/instance/db2icrt <db2_instance_name>
```
6. Change the working directory to `<Replicate_Install_Dir>/bin`.
7. Add the DB2 driver location to the Linux library path.
To do this, add the following line to the **site_arep_login.sh** file as follows:

```
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/path/lib
```

where `path` is the CLIENT path to the driver.
Example (with db2clnt1 as the instance name):

```
export LD_LIBRARY_PATH=$LD_LIBRARY_  
PATH:/home/db2clnt1/sqllib/lib:/home/db2clnt1/sqllib/lib64/gskit  
export DB2INSTANCE=db2clnt1
```
8. Optionally, confirm that the driver location was copied:


```
cat site_arep_login.sh
```
9. Add the following entry to the `odbcinst.ini` file under `/etc`:

```
[IBM DB2 ODBC DRIVER]  
Driver = /home/db2clnt1/sqllib/lib/libdb2o.so  
fileusage=1  
dontdlclose=1
```
10. Start the Replicate service:

```
./areplicate start
```
11. Optionally, confirm that the service has started:

```
./areplicate status
```

Linux data server client

1. Download **v11.5.8_linuxx64_client.tar.gz** from  [IBM Fix Central](#) and extract the files.
2. Stop the Replicate service:

```
./areplicate stop
```
3. Optionally, confirm that the service has stopped:

```
./areplicate status
```
4. Change the working directory to `/opt/client` and run `db2_insta11`.
5. Create an RSP file (see **Sample RSP file** below) and then run the following command (from `/opt/client`):

```
./db2setup -r /file-path/client.rsp
```
6. Add the following entry to the **odbcinst.ini** file under `/etc`:

```
[IBM DB2 ODBC DRIVER]
Driver = /opt/ibm/db2/v11.5/lib64/libdb2o.so
fileusage=1
dontdlclose=1
```

7. Start the Replicate service:
./areplicate start
8. Optionally, confirm that the service has started:
./areplicate status

Sample RSP file

```
*-----
* Generated response file used by the DB2 Setup wizard
* generation time: 2/18/22 4:10 PM
*-----
* Product Installation
LIC_AGREEMENT      = ACCEPT
PROD              = CLIENT
FILE              = /opt/ibm/db2/V11.5
INSTALL_TYPE      = TYPICAL
*-----
* Instance properties
*-----
INSTANCE          = inst1
inst1.TYPE        = client
* Instance-owning user
inst1.NAME        = attunity
inst1.UID         = 1000
inst1.GROUP_NAME  = attunity
inst1.HOME_DIRECTORY = /home/attunity
inst1.AUTOSTART   = NO
*-----
* Installed Languages
*-----
LANG              = EN
```

Server side ODBC setup

Bind the plan to be used for ODBC, as specified in the PLANNAME= value in the ODBC initialization file. The default name is DSNACLI. The BIND job can be found in member DSNTIJCL, which is in the SDSNSAMP library of the source DB2 installation.

Use the DB2CLI bind command to bind the ODBC-supplied packages to your intended source z/OS DB2 subsystem. This action is described in [Configuring your developer and runtime environment](#) on the IBM website. For information about the DB2CLI utility, including an example for the bind utility, see [db2cli - DB2 interactive CLI command](#) on the IBM website.

Required permissions

To enable Replicate to extract data from the source tables (Full Load and Change Data Capture), the [user specified in the IBM DB2 for z/OS endpoint settings](#) must be granted the following permissions:

- EXECUTE on the IFI reading the UDTF (only required for Change Data Capture)
- SELECT on the source tables and on the following system catalog tables:
 - SYSIBM.SYSDUMMY1
 - SYSIBM.SYSTABLES
 - SYSIBM.SYSTABLESPACE
 - SYSIBM.SYSPARMS
 - SYSIBM.SYSROUTINES
 - SYSIBM.SYSCOLUMNS
 - SYSIBM.SYSLGRNX
 - SYSIBM.SYSTABLEPART
- MONITOR2 to be able to start IFI sessions (only required for Change Data Capture)

Change data capture requirements

To capture changes from IBM DB2 for z/OS, Replicate uses a special program - invoked as an external routine - which is a user-defined table function (UDTF). Both this program (a load module) and the UDTF need to be installed and configured on the z/OS system before changes can be captured. The installation procedure, which should be performed by the DBA, is described in see *Installing and configuring R4Z (page 290)*.

Additionally, the DATA CAPTURE CHANGES attribute must be set for every table whose changes you want to capture. You can either do this manually or allow Replicate to do this by leaving the **Automatically enable DATA CAPTURE CHANGES (requires admin privilege)** option enabled (the default) in the endpoint settings.

Limitations and considerations

You should be aware of the following limitations and considerations when using an IBM DB2 for z/OS source:

- Capturing changes from Catalog Tables is not supported.
- During a task with Full Load and Apply Changes enabled, duplicate keys may be replicated to the target. This is because records that were updated and cached during the Full Load stage of the task may have the same timestamp as the original records that were inserted during Full Load. Note that this usually only happens when the time difference between the inserted record and the updated (cached) record is very small (milliseconds).
- The "enableWLB" driver option is not supported. To force the connection to automatically switch to another LPAR in a parallel sysplex (i.e. "failover"), set "enableACR" to "true" and "enableWLB" to "false" in the driver configuration.
- Replication of Views is supported during Full Load only (i.e. CDC is not supported).
- Capturing changes from an environment with mixed RBA entries (6 and 10 byte formats) is not supported.
- Replicating data with 4-byte emoji characters is not supported.

Handling actions resulting in subtype 83

Subtype 83 is a special diagnostic log record. The following topic describes how Replicate handles actions resulting in subtype 83.

Subtype 83 commands

Action	How it is Handled by Replicate
ALTER TABLE (With some variations, e.g. ADD COLUMN)	Handled without issue.
MODIFY RECOVERY (With some variations)	Handled without issue.
LOAD REPLACE (All variations)	A warning message will be generated, with the following text (example): DB2z utility (subtype 83) variation 4 (LOAD REPLACE) at LSN=0000000000123abc0000 Additionally, the table will be suspended if the internal Replicate parameter "db2LoadOption" is set to "SUSPEND". The default is "IGNORE", in which case a warning message will still be printed.
LOAD RESUME YES (All variations except SHRLEVEL CHANGE)	A warning message will be generated, with the following text (example): DB2z utility (subtype 83) variation 3 (LOAD RESUME YES) at LSN=0000000000123abc0000 Additionally, the table will be suspended if the internal Replicate parameter "db2LoadOption" is set to "SUSPEND". The default is "IGNORE", in which case a warning message will still be printed.
REORG DISCARD (If any rows are discarded)	A warning message will be generated, with the following text (example): DB2z utility (subtype 83) variation 5 (REORG DISCARD) at LSN=0000000000123abc0000 Additionally, the table will be suspended if the internal Replicate parameter "db2LoadOption" is set to "SUSPEND". The default is "IGNORE", in which case a warning message will still be printed.

Action	How it is Handled by Replicate
CHECK DATA DELETE (If any rows are deleted)	<p>A warning message will be generated, with the following text (example):</p> <p>DB2z utility (subtype 83) variation 6 (CHECK DATA DELETE) at LSN=0000000000123abc0000</p> <p>Additionally, the table will be suspended if the internal Replicate parameter "db2LoadOption" is set to "SUSPEND". The default is "IGNORE", in which case a warning message will still be printed.</p>
RECOVER TO POINT IN TIME	<p>A warning message will be generated, with the following text (example):</p> <p>DB2z utility (subtype 83) variation 7 (RECOVER TO POINT IN TIME) at LSN=0000000000123abc0000</p> <p>Additionally the table will be suspended if the internal Replicate parameter "db2LoadOption" is set to "SUSPEND". The default is "IGNORE", in which case a warning message will still be printed.</p>



It's possible that LOAD RESUME NO, without REPLACE, may also create the subtype 83 log record. In such a case, the warning message shown above would indicate an unknown variation number, with (UNIDENTIFIED) as the description. All other actions would be the same as described above.

It is important to note that data loaded by this utility variation will not be captured. Consequently, this utility variation should be avoided.

Supported data types

The following table shows the IBM DB2 for z/OS database source data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.


For information on how to view the data type that is mapped in the target, see the section for the target database you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

Supported data types

IBM DB2 for z/OS Source Data Types	Qlik Replicate Data Types
INTEGER	INT4
SMALLINT	INT2
BIGINT	INT8

9 Adding and managing source endpoints

IBM DB2 for z/OS Source Data Types	Qlik Replicate Data Types
DECIMAL (p,s)	NUMERIC (p,s) <div style="border: 1px solid gray; padding: 10px; margin-top: 10px;">  <p><i>If a decimal point is set to a comma (,) in the DB2 configuration, you will need to configure Replicate to support the DB2 setting.</i></p> <p><i>For information on how to configure Replicate, see R4Z Configuration Dependency on Host.</i></p> </div>
FLOAT (8)	REAL8
DOUBLE	REAL8
REAL	REAL4
DECFLOAT (p)	If precision = 16, then: REAL8 If precision = 34, then: STRING
GRAPHIC	If n<=127, then: WSTRING
VARGRAPHIC	If n<=16k double byte chars, then: WSTRING
LONG VARGRAPHIC	CLOB
CHAR (n)	STRING n<=255
VARCHAR (n)	STRING n<=32k
LONG VARCHAR (n)	CLOB n<=32k
CHAR (n) FOR BIT DATA	BYTES
VARCHAR (n) FOR BIT DATA	CLOB
LONG VARCHAR FOR BIT DATA	BLOB
DATE	DATE

IBM DB2 for z/OS Source Data Types	Qlik Replicate Data Types
TIME	TIME
TIMESTAMP	DATETIME (6)
BLOB	BLOB
CLOB	CLOB Maximum size: 2 GB
DBCLOB	CLOB Maximum size: 1 G double byte chars
XML	CLOB
BINARY	BYTES
VARBINARY	BYTES
ROWID	IGNORED
TIMESTAMP WITH TIME ZONE	NOT SUPPORTED

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.



You can also use IBM DB2 for z/OS File Channel files as a source. For more information, see *Using the Qlik Replicate file channel (page 1061)*.

To add an IBM DB2 for z/OS source endpoint to Qlik Replicate:

1. In the Qlik Replicate Console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Microsoft SQL Server database. This is optional.
4. Select **Source** as the database **role**.
5. Select **IBM DB2 for z/OS** as the database **Type**.



When this endpoint is used as a duplicated source in a Log Stream Staging setup, select the **Read changes from log stream** check box and then select the relevant Log Stream Staging task from the drop-down list.

For information on setting up and managing Log Stream Staging tasks, see *Using the Log Stream* (page 1072).

6. Choose one of the following:

- **Use database alias (DSN)** - Specify the database alias you defined using the DB2 Configuration Assistant or the DB2 Command Line Processor.
- **Use these connection properties** - If you choose this option, enter the IBM DB2 for z/OS **Server** (hostname or IP address), **Port** and **Location** in the designated fields.



If the **Server** is a parallel SysPlex and data sharing members reside on multiple LPARs, the host address may specify a DVIPA - a Dynamic Virtual IP Address (to utilize the system redundancy and load-balancing in Replicate processing). In this case, the port numbers must be identical for all members.

The **Location** should be the DB2 location name defined during the installation. This should be a relational database management system under z/OS, which can be either a subsystem or a group connection. This is the logical name which serves applications in order to designate resources managed by this system, either using the SQL CONNECT instruction, or placing it as a qualifier of a table (preceding the schema name).

To see the location name, use the “-DIS DDF” DB2 command (option 7 under the DB2I panel in ISPF), or look in message DSNL004I in the job log of the <ssid>MSTR address space.

7. Enter the **User name** and **Password** for an authorized user of the specified IBM DB2 for z/OS database. For a list of the permissions that need to be granted to this user, see *Required permissions* (page 280). If you do not know this information, consult with your IBM DB2 for z/OS database administrator (DBA).



- *This information is case sensitive*
- *Make sure that the specified user has the required access privileges*

8. **Provider:** Leave the default unless it was changed during the driver installation. Note that this should be the same as the name specified in the ODBC Data Source Administrator.

Setting advanced connection properties

In the **Advanced** tab, you can set advanced connection properties such as overriding CCSID to Character Set mapping, setting Change Data Capture properties, and setting internal Replicate parameters.

Overriding CCSID to character set mapping

In some cases, character data in source tables may be encoded in a different CCSID than what is declared in the source database table definition. For example, a specific table or column definition might indicate that it uses CCSID 500 (EBCDIC International) whereas in fact, it uses CCSID 1148 (ENCODIC International with EURO).

9 Adding and managing source endpoints

In this case, you can tell Replicate that the source definition CCSID 500 should be treated as CCSID 1148 (specifically, the character set named IBM-1148).

Note that when the source table definition specifies CCISD 65535 (meaning character set is unknown), you must specify what character set should be assumed when reading data from that table or column.



If there is a conflict between the character set mapping for a specific column and the character set mapping defined in the endpoint settings, the column-level character set mapping takes precedence.

For more information on overriding character set mapping at column level, see [Using the Transform tab \(page 1086\)](#).

To do this:

1. In the **Override CCSID to Character Set Mapping** section, click the **New** button.
A row is added to the table.
2. Enter the CCSID in the **CCSID** column and the code page in the **Character set** column.
The **Character set** (code page) must be specified in the proper format (e.g. ibm-1148_P100-1997). For a list of valid formats, see [the ICU Converter Explorer](#).
3. Repeat to map additional CCSID values.

Converting to a custom code page

Perform the following procedure if your source endpoint tables are defined with an incorrect CCSID and the correct definition is actually in a UCM file.

1. Create a mapping data file with the file extension **.ucm**.



If you edit an existing UCM file, you must also change the values of the `<code_set_name>` and `<icu:alias>` properties. If the file does not contain an `<icu:alias>` property, then you only need to change the value of the `<code_set_name>` property.

2. Create a CNV file for the UCM file by running the following command:

```
<product_dir>\bin\makeconv.exe -v <file_name>.ucm
```

Example:

```
"c:\Program Files\Attunity\Replicate\bin\makeconv.exe" -v 1047_EX.ucm
```

This will create a CNV file with the same name as the UCM file (for example, `1047_EX.cnv`).

3. Create a new subfolder named **icudt<XX>l** under `<product_dir>\bin` where the **XX** is the same as the number in the **icudt<XX>.dll** file name.
For example, If the DLL file name is **icudt69.dll**, create a new subfolder named **icudt69l**.
4. Copy the CNV file to the subfolder you just created (`<product_dir>\bin\icudt69l` in the example above).



When using the Replicate File Channel, the file should be copied to the same location on both Replicate servers.

5. Add a new character set mapping as follows:
 - a. In **CCSID** column, enter the original source CCSID number (e.g. **1047**)
 - b. In the **Character set column**, enter the name of the CNV file without the extension (e.g. **1047_EX**).
6. Restart the Qlik Replicate UI Server service.

Change data capture properties

- **Check for changes every:** How often to check for new changes when the database is quiet. When the database is active, changes are captured as soon as they are detected.
- **CDC reader UDTF name:** The name of the Qlik-supplied User-Defined Table Function, which is used for CDC. Specify the two-part name using the values you have chosen for schema name, **&R4ZSCNM**, and the function name, **&R4ZIFITF**.
- **UDTF result set size (MB):**

Specify the maximum size to be accumulated by result rows returned by the Qlik-supplied User-Defined Table Function, through CDC tasks using this endpoint. A larger result set will cause less overhead time establishing the IFI session, but will result in greater memory consumption by DB2. Specifically, because the result set is a LOB, you might need to increase the LOBVALA limit in the DB2 configuration (DSNZPARM), specifying the maximum LOB size allowed per user. Assuming all CDC tasks are configured with the same ODBC user-ID, to accommodate for all tasks running concurrently, set LOBVALA to $2 \times 1024 \times$ the following size:

```
SUM[(i=all endpoints) : resultset size in endpoint(i) * #_tasks using endpoint(i)]
```

(Resultset size is in Mbytes, so it needs to be multiplied by 1024 to reflect LOBVALA size, which is in Kbytes).

Another DSNZPARM limit, LOBVALS, specifies the total size when across all user-IDs; so, if there are multiple user IDs, you need to monitor this value as well. Note, however, that LOBVALS is specified in Mbytes, unlike LOBVALA.
- **Automatically enable Data Capture Changes (requires admin privileges):** For Qlik Replicate to be able to capture changes, the Data Capture Changes attribute needs to be set on all relevant source tables. You can either do this manually or allow Replicate to perform this action by leaving this option enabled (the default). When this option is enabled, the connecting user must have ALTER permission on the source tables being captured.

Setting internal parameters

Internal parameters are parameters that are not exposed in the UI. With the exception of the parameters listed in *Authorized internal parameters (page 288)* at the end of this section, you should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
The parameter is added to the table below the search box with its default value.
3. Change the default value as required.

9 Adding and managing source endpoints

4. To add more parameters, repeat steps 2 and 3 above.
5. To reset a parameter value to its default, click the "Restore default value" icon at the end of the row.

Authorized internal parameters

The following table provides a list of internal parameters that can be used as required.

Internal parameters

Parameter name	Values	Default; Max; Min.
ifi306MessageLevel	0 – No DEBUG traces; 1 – include DEBUG traces; 9 – full DEBUG	0; 0; 9.
ifi306BufferSize	Size (in bytes) of Log data buffer used for IFI reading	65536; 65536; 1048576.
additionalConnectionProperties	<keyword>=<value>;... Example: cursorhold=1;patch2=15	(empty string)

R4Z configuration dependency on host

This table in this section provides a list of R4Z configuration values, which can be set during installation or as part of the CDC service setup, or as part of the endpoint definition.

R4Z configuration values

R4Z Topic	Depends upon Property	Defined at/Inquired by	Resolution
Total Size of R4Z ECSA structures	Available ECSA size	/DNET,CSM,... OWNERID=ALL	Make sure ECSA size is not exceeded.
MAXRESULTSETSECS parameter in the CDC service	Maximum time DB2 thread waits for a locked resource	DSNZPARM IRLMRWT param. in SDSNSAMP (DSNTIJUZ)	Set at least two seconds below the SSID spec.
MAXRESULTSETSIZEparameter in the CDC service	Maximum LOB size available per user-ID	DSNZPARM LOBVALA param. in SDSNSAMP (DSNTIJUZ)	2 * (resultset size * max. session)

9 Adding and managing source endpoints

Decimal point: period or comma	Designates the decimal editing separator	DSNHDECM DECIMAL= param. In DSNTIJUZ	If the decimal separator is a comma, set the additionalConnectionProperties internal parameter to patch2=15.
--------------------------------	--	---	--

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

Sample XMIT files “Receive” job

The following is a sample job for receiving the LOAD and INSTALL libraries.

```
//*****  
//*                               Sample JCL for receiving                *  
//* IBM DB2 for z/OS endpoint installation kit      *  
//* 1. Add a jobcard                               *  
//* 2. Replace all <xmit-HLQ> with the High Level Qualifier used      *  
//*    for receive files                               *  
//* 3. Replace all <r4z-vn-hlq> with the High Level Qualifier chosen *  
//*    for the installation files                               *  
//*****  
//RCVLOAD EXEC PGM=IKJEFT01,DYNAMNBR=20  
//SYSPRINT DD SYSOUT=*  
//SYSTSPRT DD SYSOUT=*  
//SYSTSIN DD *  
RECEIVE INDSN('<xmit-HLQ>.LOAD.XMIT') NODISPLAY  
DATASET('<r4z-vn-hlq>.LOAD')  
/*  
//RCVINSTL EXEC PGM=IKJEFT01,DYNAMNBR=20  
//SYSPRINT DD SYSOUT=*  
//SYSTSPRT DD SYSOUT=*  
//SYSTSIN DD *  
RECEIVE INDSN('<xmit-HLQ>.INSTALL.XMIT') NODISPLAY
```

```
DATASET(' <r4z-vnd-h1q>.INSTALL ')  
/*
```

Installing and configuring R4Z

This Qlik R4Z component is required to replicate data from IBM DB2 for z/OS.

This section describes how to install, upgrade, and configure R4Z in the DB2 mainframe environment.

In this section:

- *R4Z components and the associated environment (page 290)*
- *R4Z product libraries (page 290)*
- *Receiving product-supplied modules and scripts (page 291)*
- *Installing and upgrading the R4Z product in z/OS (page 293)*
- *Controlling the CDC process (page 302)*

R4Z components and the associated environment

The relationship between all components involved in the replication process is described below.

- The IBM DB2 for z/OS endpoint is associated with:
 - A z/OS host, which can be a single system (LPAR) or a Sysplex, via a TCP/IP address. Referring to a Sysplex requires defining a VIPA that points to a set of systems.
 - A DB2 location name active within the above z/OS host, which is the logical identification of a DB2 subsystem ID (non-data sharing) or a “group attach”, which may refer to several members of a data sharing DB2.

The endpoint also allows you to set a **CDC reader UDTF name**. This is an external UDTF (User-Defined Table Function) which is referenced by a “SELECT ... FROM TABLE(...)” statement issued from the replication task. The CDC reader UDTF is created in DB2, at the location name specified above, by SQL DDL, as part of the installation. This definition also specifies:

- The name of the external program to be executed for serving the function call. This name must not be changed.
- The name of the WLM APPL ENV (application environment) through which the program is to be executed. This is defined as part of the installation. The WLM APPL ENV specifies, as one of its attributes, the name of a JCL procedure for initiating a started task (STC) address space. Such STC address spaces (also known as WLM servers) are started and managed by WLM upon calls to the UDTF, to invoke the external program to be executed for serving the function call. The JCL procedure is stored in a PROCLIB library as part of the installation.
- The **CDC reader UDTF name**: If this name matches a specific pattern, it specifies a CDC service qualifier to be associated with the UDTF invocation. The CDC service, which is defined in the R4Z CONFIG library as part of the installation and may be altered later, designates certain system resources and configuration settings to be used during CDC processing. If the UDTF name does qualify for the said pattern, a default CDC service is associated with the UDTF.

R4Z product libraries

The following library types are required by the product:

- **LOAD** and **INSTALL** libraries - These libraries are received from the installation kit. Therefore, their content is version-dependent and cannot be reused by other versions.
- **CONFIG** and **CNTL** libraries - These libraries are created during the installation and configuration. As a result, their content can be reused by successive versions of the product (depending on the specific case).

Examples of **LOAD** and **INSTALL** library file names:

- **LOAD:** r4z.v2022r5.sp0.load.xmit
- **INSTALL:** r4z.v2022r5.sp0.install.xmit

To accommodate for these different use types, it is recommended to use separate high-level qualifiers for the R4Z product libraries:

- R4Z "version-specific" high-level qualifier, with the following libraries:
 - **LOAD** - Found in the kit; contains executable load modules.
 - **INSTALL** - Found in the kit; contains installation scripts/JCLs and sample jobs.
This qualifier will henceforth be referred to as **<r4z-vnd-h1q>**.
- R4Z "basic" high-level qualifier, independent of versions, with the following libraries:
 - **CONFIG** - Created during the installation; contains the CDC services configuration.
 - **CNTL** - To be created by the person performing the installation, for saving JCL components - jobs and Include members. It is possible to allocate several CNTL libraries, assigning them a name that contains the CDC service name. This is useful if the CDC services are to be managed independently of one another.
This qualifier will be henceforth referred to as **<r4z-bsc-h1q>**.

Receiving product-supplied modules and scripts

The following instructions explain how to download and receive the required modules and scripts.

To receive the load modules and installation components:

1. Download R4Z as described in *Prerequisites (page 277)*.
2. Unzip the file contents to your workstation.

The zip file contains the following files:

- r4z.vxx.ryy.spzz.install.xmit - JCLs; ISPF dialog components.



The version identifiers, namely; major release, minor release, and service pack, are indicated by xx, yy and zz respectively.

- r4z.vxx.ryy.spzz.load.xmit - Executable load modules.
- version.txt - A text file listing the version identifiers in this kit.
- r4zrrcv.txt - A sample job for populating the **INSTALL** and **LOAD** libraries from the matching XMIT files.

3. Pre-allocate the following 1 cylinder datasets with LRECL=80,RECFM=FB,DSORG=PS:

- <xmit-HLQ>.INSTALL
 - <xmit-HLQ>.LOAD
4. Perform a binary transfer of `r4z.vxx.ryy.spzz.load.xmit` to **<xmit-HLQ>.LOAD** and of `r4z.vxx.ryy.spzz.install.xmit` to **<xmit-HLQ>.INSTALL**.
 5. In TSO, use the RECEIVE command to create and populate the INSTALL and LOAD libraries from the matching xmit files, which should now exist in the host z/OS.

You can do that in either of the following methods:

- Use the sample job provided in **r4zrrcv.txt** (see below) to invoke the RECEIVE command.
- Run the RECEIVE command directly under TSO:
 - a. From the ISPF TSO command shell run the command below per each library <type>, INSTALL or LOAD:

```
Receive indsn('<xmit-HLQ>.<type>')
```
 - b. When prompted INMR906A Enter restore parameters or 'DELETE' or 'END' +, respond with the output dataset and other parameters; for example:

```
Receive dsn('<r4z-vnd-HLQ>.<type>').
```

These RECEIVE commands will populate the INSTALL and LOAD libraries.

Sample Job

Below is the sample RECEIVE job found in the **r4zrrcv.txt** file. For non-SMS datasets, change <volser> to a valid volume serial number. For SMS datasets, delete **UNIT** and **VOLUME**.

```
//RECEIVE JOB NOTIFY=&SYSUID  
  
//RECEIVE1 EXEC PGM=IKJEFT01,DYNAMNBR=11  
  
//SYSTSPRT DD SYSOUT=*  
  
//SYSTSIN DD *  
  
RECEIVE INDSNAME(<xmit-HLQ>.INSTALL')  
  
DSNAME('<r4z-vnd-HLQ>.INSTALL') UNIT(3390) VOLUME(<volser>)  
  
RECEIVE INDSNAME('<xmit-HLQ>.LOAD')  
  
DSNAME('<r4z-vnd-HLQ>.LOAD') UNIT(3390) VOLUME(<volser>)
```

Installing and upgrading the R4Z product in z/OS

This section explains how to install and upgrade the R4Z product in z/OS.



*The Replicate and R4Z versions you install must be part of the same release, which can either be an initial release or a patch/service release. This means that if you upgrade Replicate, you must also upgrade your R4Z installation (to the version that correlates with the Replicate release you are installing), unless it has the same build number as your currently installed R4Z. Note that despite being part of the same release, the Replicate and R4Z build numbers (reflected in their file names) might not be the same. For example, Replicate May 2022 SR2 includes the following files: **QlikReplicate_2022.5.0.499_X64.zip** (the Replicate installer) and **QlikReplicate_2022.5.0.290_r4z.zip** (the R4Z installer).*

Installing the R4Z product in z/OS

The configuration steps below include submitting jobs found in the INSTALL library (referred to as <**r4z-vnd-hlq**>. **INSTALL** in *R4Z product libraries* (page 290)). Each job requires the INSTALL library (or its copy) as its JCLLIB, and must have both a job card and JES control statements that comply with the site's regulations.

Applying the installation in the processing environment

During the installation process you are required to submit most of the members with the prefixed "DO*" in the library. Before you submit these members you have to edit them as follows:

1. Set the job name, the job card parameters, and JES control statements to comply with your system requirements.
2. Set the library specified in the JCLLIB ORDER parameter as the INSTALL library.

9 Adding and managing source endpoints

Therefore, before submitting any job, you should create a JCL library that contains all updated “DO” jobs. If you want to create a single set of jobs per all CDC services, it is advisable to assign the JCL library the name `<r4z-hlq>.COMMON.CNTL`; otherwise use the CDC service qualifier as a DSNAMES node; for example `<r4z-bsc-hlq>.<CDC-qual>.CNTL`.



The installation kit also includes members with the prefix “UN*”; each job starting with “UN” performs the UNDO action of the job by the same name that starts with “DO”. If need to use one of these jobs, make the same changes that you applied to the “DO” jobs.

Step 1: Setting the INSTALL library INCLUDE members

You need to set the JCL symbols below before running any job, as the “DO” jobs use these JCL symbols to configure the R4Z processing. To change their values, you will need to first run the “UNDO” job with the old value, and then the “DO” job with the new value.

Variables common to the entire R4Z installation

DFSYMLST	INCMEM: define general system resources	Once per installation
Sets: &SSID, &PROCLIB, &DB2HLVL, &CEERUNL, &DB2LOADL, &DB2EXITL, &DB2RUNL, &DB2VER, &R4ZHLVL, &R4ZLOADL, &R4ZINSTL, &R4ZCNFGL, &R4ZUSER, &DEFWLMAE, &ALCSMS, &ALCSMSPM, &ALCVLSER		

Edit this member to set the JCL symbols to be used by jobs in this library and then save (override) the member. These symbols are listed in the *Installation symbols (page 299)* table below.

Here is a more detailed explanation of several of the parameters:

- Parameters that specify the R4Z product library names:
 - &R4ZHLVL - Set to the high-level-qualifier which is common to the R4Z INSTALL, LOAD and CONFIG libraries, i.e. the longest common part of these libraries names:
 - &R4ZVNDH - Set to the value you used as the “versioned” high-level qualifier (`<r4z_vnd_HLQ>`) when receiving the INSTALL and LOAD libraries.
 - &R4ZBSCH - Set to the value you chose for the “basic” high-level qualifier (`<r4z_bsc_HLQ>`), which will be used when creating the CONFIG library.
 - &R4ZINSTL, &R4ZLOADL, &R4ZCNFGL - Specify these only if their value is different than that set within DFSYMLST member, based on the symbols described above.
- Parameters that control DASD allocation:
 - ALCSMSPM, &ALCVLSER - Use these parameters to designate the volume(s) on which the CONFIG library, and some intermediate datasets will be allocated. Use &ALCSMSPM to specify SMS allocation attributes in a comma-delimited, space-terminated string, in the following format:


```
' [STORCLAS=...] [,MGMTCLAS=...] [,DATACLAS=...] '
```

Use `&ALCVLSER` to specify the volume serial.

If SMS-managed allocation is driven by ACS routines, you can omit all attributes and specify `&ALCSMSPM='` (a single space) instead.

If you specify `&ALCSMSPM='` (a single space), you must specify a value for the `&ALCVLSER` parameter.

For non SMS-managed allocations, use `&ALCVLSER` to specify the volume serial.

- CDC service and UDF- related variables:

DFSYMPER	Define names for the UDF, the WLM application env and STC procedure, based on the CDC service qualifier	Per R4Z installation
Uses: <code>&CDCSRV</code> Sets: <code>&R4ZWLMPR,</code> <code>&R4ZAPPNV,</code> <code>&R4ZIFITF,</code> <code>&R4ZSCNM,</code> <code>&STCUSER,</code> <code>&UDFUSER</code>		
DFSYMONE	Define names for the UDF, the WLM application env and STC procedure, assuming a single CDC service is used	Per DB2 subsystem
Sets: <code>&R4ZWLMPR,</code> <code>&R4ZAPPNV,</code> <code>&R4ZIFITF,</code> <code>&R4ZSCNM,</code> <code>&STCUSER,</code> <code>&UDFUSER</code>		

Depending on the value selected for `DEFWLMAE`, `ONE` or `PER`, edit member `DFSYMONE` or `DFSYMPER` respectively, and select values for the WLM `APPLENV`, the WLM JCL procedure for the CDC reader `UDTF`.



In `DFSYMPER` member, these names are formed by a symbol designating the name prefix, followed by a 4-character qualifier specifying the CDC service. Do not remove the double-underscore at the end of the `APPLENV` name or `UDTF` name – these are used as delimiters, denoting the CDC service to be associated with the named resource.

Step 2: Performing configuration actions

1. Set the R4Z LOAD library as APF-authorized.

To set a library as APF-authorized temporarily (until the next IPL or APF update), you can run one of the following system commands:

- For non SMS-managed data sets:
`setprog apf,add,dsn=<xmit-HLQ>.LOAD,volume=<volser>`
- For SMS-managed data sets:
`setprog apf,add,dsn=(<xmit-HLQ>.LOAD,sms`
To make the setting permanent, add the library to your `PROGxx` member in `PARMLIB`.

9 Adding and managing source endpoints

2. Set up a WLM application environment to execute the CDC process. |

DO2WLMAE	Define WLM Application environment	Per CDC service
Accepts: &CDCSRV Uses: &R4ZAPPNV, &R4ZWLMPR, &CEERUNL, &DB2EXITL, &DB2LOADL, &DB2RUNL		
DO2WLMPR	Create JCL procedure for WLM STC	per WLM procedure
Accepts: &CDCSRV Uses: &PROCLIB, &R4ZWLMPR, &SSID		
DO2WLMPV	Job: In define WLM server as a resource in RACF, and authorize APPLENV and DB2 subsystem to use it	per CDC service and WLM procedure
Accepts: &CDCSRV Uses: &R4ZAPPNV, &R4ZWLMPR, &STCUSER		

Depending on the value set for DEFALMAE, run the following jobs, once (if ONE is selected) or PER each CDC service to be defined (if PER is selected), after setting JCL symbol CDCSRV with the relevant CDC service qualifier.

- a. Edit job DO2WLMAE; set symbol &CDCSRV to the relevant CDC service qualifier and submit. The job will define the WLM APPLENV in the WLM policy dataset and activate it.
 - **Prerequisites:**
 - &R4ZWLMPR is the name of the started task that WLM will use. You need to associate a valid user to the started task and set &STCUSER to that user.
 - The submitter needs UPDATE access to MVSADMIN.WLM.POLICY, to query and update the WLM policy.
 - UPDATE on the procedure library specified in DSSYMLST in the &PROCLIB symbol.
- b. Edit job DO2WLMPR; set the symbol &CDCSRV to the relevant CDC service qualifier and submit. The job will create a JCL procedure for the APPLENV, and should terminate with RC 0.
 - **Authorizations:**
UPDATE on the PROCLIB library.
- c. Edit job DO2WLMPV; set the symbol &CDCSRV to the relevant CDC service qualifier and submit. The job will set RACF privileges for the APPLENV to use the WLM server, and for the DB2 subsystem to call it. The job should terminate with RC 0.
 - **Authorizations:**
ALTER on class SERVER
If you are using a security package other than RACF (for example, as CA-TSS or CA-ACF2), you will need to run the commands relevant to your security package.

3. Create the CDC reader UDTF(s).

DO3SRVDF	Create the default CDC service UDTF	Per DB2 subsystem
Uses: &R4ZSCNM, &R4ZIFITF, &R4ZCNFGL, &ALCSMS, &ALCSMSPM, &ALCVLSER, &SSID, &DB2EXITL, &DB2LOADL, &DB2RUNL, &UDFUSER		
DO3SRVTF	Create a UDTF to be used by a specific CDC service	Per CDC service and DB2 subsystem
Accepts: &CDCSR Uses: &R4ZSCNM, &R4ZIFITF, &R4ZCNFGL, &ALCSMS, &ALCSMSPM, &ALCVLSER, &SSID, &DB2EXITL, &DB2LOADL, &DB2RUNL, &UDFUSER		

The first job creates the default-service UDTF with specific name &R4ZIFITF (under schema name &R4ZSCNM) and grants execution on it to &UDFUSER. It also instantiates a “supplied” CDC service, CDC1 (by creating such a member CONFIG library), setting it with default configurations. This job must be run first.

The second job, which is ran per each CDC service defined explicitly, performs exactly as the first job, except each UDF is associated with a specific CDC service, which is instantiated from the CDC1 service.

- a. Run job DO3SRVDF. It will:
 - Create a UDTF with an un-suffixed name, which will be associated with the default CDC service (CDC1).
 - Create the CONFIG library and place a member named CDCS, to be used when no CDC service-related member exists.

Jobs should terminate with RC4.

- b. Per each CDC service you want to define, edit job DO3SRVTF; set symbol &CDCSR to the relevant CDC Service qualifier and submit. It will:
 - Create a UDTF with its name suffixed "___&CDCSR", which will be associated with the default CDC service ""&CDCSR".
 - Create a member named "&CDCSR" in the CONFIG library.
 - Set JCL symbol CDCSRV with the relevant CDC service qualifier (xxxx), and submit the job.

Jobs should terminate with RC4.

- **Authorizations:**
The submitting user must have one of the following authorizations:
 - SYSADM or SYSCTRL authority.
 - CREATEIN authorization in the schema designated by &R4ZSCNM.

4. Enable specified user-ID to run CDC tasks.

9 Adding and managing source endpoints

DO4GRANT	Grant ODBC user access to catalog tables, and permission to read CDC	Per DB2 subsystem
Uses: &R4ZUSER, &SSID, &DB2EXITL, &DB2LOADL, &DB2RUNLR		

This job enables the user-ID selected for ODBC interaction. This is the “ODBC user”, that is authorized (DB2-wise) to read CDC data and some of the DB2 catalog tables.



There is no point submitting the job from the ODBC user itself, (as a user cannot grant itself privileges), or from a user that does not own the privileges it attempts to grant.

Run job DO4GRANT. It will:

- GRANT SELECT to the ODBC user on the following tables:
 - SYSIBM.SYSTABLES
 - SYSIBM.SYSTABLESPACE
 - SYSIBM.SYSCOLUMNS
 - SYSIBM.SYSROUTINES
 - SYSIBM.SYSDUMMY1
 - ALTER SYSIBM.SYSTABLES enabling DATA CAPTURE.

Jobs should terminate with RC 0 or RC 4.

Authorizations:

The submitting user must have either of the following authorizations:

- SYSADM or SYSCTRL authority.
- TRACE/ MONITOR2 with grant option.

5. Provide security permissions for the WLM started task to cancel itself. In RACF this is done as follows:

- a. Define a PROFILE for the OPERCMDS class called PERMIT MVS.CANCEL.STC.mbrname.*:

```
RDEFINE OPERCMDS MVS.CANCEL.STC.mbrname.* UACC(NONE) SETROPTS  
REFRESH RACLIST(OPERCMDS)
```

Where `mbrname` is the name of the started task.

- b. Grant the USERID assigned to the WLM started TASK the authority to cancel the started task:

```
PERMIT MVS.CANCEL.STC.mbrname.* CL(OPERCMDS) ACCESS(UPDATE) ID  
(stcuser)
```

Where `stcuser` is the USERID associated with the WLM started task.

6. Performance considerations:

- a. **Full Load** – The Full Load processes will utilize any available ZIIP processors. Best practice is to set the performance goal that the processes get classified to lower than online workloads. In a very busy system, you might want to consider setting **Honor Priority** to **NO** for the service class assigned to the Full Load. This requires z/OS 2.1 + OA50845 or z/OS 2.2 and above.
- b. **CDC** – Best practice is to set the performance goal for the CDC processes equal to any online workloads and higher than the Full Load goal. These processes can either be classified using any of the following:
 - The name of the WLM application environment started task
 - The DDF classification with 'Procedure Name'

9 Adding and managing source endpoints

Installation symbols

Symbol Name	Description	Attribute <hr/> Default Value
&SSID	The subsystem-ID of the source DB2.	4-character name.
DSN1	High-level qualifier of DB2 installation libraries, not including SSID qualifier.	<qualifier>.<qualifier>...
&DB2HLVL		
DSNB10	High-level qualifier of DB2 SDSNLOAD library.	Library name.
&DB2LOADL		
&DB2HLVL..&SSID..SDSNLOAD	High-level qualifier of DB2 SDSNEXIT library.	Library name.
&DB2EXITL		
&DB2HLVL..&SSID..SDSNEXIT	High-level qualifier of DB2 RUNLIB.LOAD library.	Library name.
&DB2RUNL		
&DB2HLVL..&SSID..RUNLIB.LOAD	The z/OS DB2 version.	Number (10..12)
&DB2VER		
11	CDC service: A set of resources - ECSA structures, a table function and possibly a WLM application environment - used for reading CDC data by all tasks designating this service.	Qualifier of up to four characters, forming a valid name.
&CDCSRV		
CDC1	z/OS LE (language environment) and C++ runtime library.	Library name.
&CEERUNL		

9 Adding and managing source endpoints

Symbol Name	Description	Attribute Default Value
CEE . SCEERUN		
&PROCLIB	JCL procedure library where third-party JCL procedures are located. You can choose a PROCLIB used by third-party products.	Library name.
N/A		
&R4ZHLVL	R4Z high-level qualifier.	<qualifier>.<qualifier>...
R4Z		
&DEFWLMAE	Determine whether a separate WLM APPL ENV is to be created PER CDC service, or all services are to use ONE WLM APPL ENV	'PER' or 'ONE'
N/A		
&R4ZLOADL	R4Z load library.	Number (09..12).
&R4ZHLVL . . LOAD		
&R4ZWLMR	WLM STC for the Replicate application environment.	Member name (JCL procedure).
&SSID . WR4Z		
&R4ZAPPNV	The WLM application environment serving the Replicate UDTF.	Upper case name, up to 32 characters (underscores allowed).
R4ZWLM		
&STCUSER	The user ID assigned to regions started by WLM application-environment	User ID (up to 7 characters).

9 Adding and managing source endpoints

Symbol Name	Description	Attribute Default Value
(none)		
&R4ZUSER	User ID used for issuing SQL select statements IFI306 READS; may be overridden by &UDFUSER.	User ID (up to 7 characters).
ATTUSER		
&R4ZSCNM	Schema name qualifying the UDTF routine (to be specified in the IBM DB2 for z/OS endpoint settings)	Schema name.
&R4ZUSER		
&R4ZIFITF	User-defined function name (to be specified in the IBM DB2 for z/OS endpoint settings).	Function name (not including the schema name).
R4ZUDF		

Migrating the R4Z product in z/OS



If an older version of the product is already installed and operating in the environment, it is mandatory to complete these instructions before starting the installation of the new version.

If you want to upgrade your R4Z installation and override the current installation, or if you want to upgrade your R4Z installation without overriding the currently installed version, you need to perform the relevant steps outlined below.



If the existing version is 5.5 or 6.0, consult with Qlik Support. Depending on the level, running the two versions in parallel may not be supported.

To upgrade your R4Z installation without overriding the currently installed version:

If you are running version 6.1 or above, you should be able to run the two versions in parallel, provided that:

- Each version has its own WLM server JCL procedure, where the LOAD library of that version is specified in the STEPLIB DD of the JCL procedure.
- The R4Z product must already be installed with DEFWLM=PER specified, to ensure that each CDC service is executed in its own WLM server.

To upgrade your R4Z installation and override the currently installed version, suspend all CDC services using R4Z as follows:

1. Use the following operator command to quiesce the WLM application environment and prevent clients from auto-activating the CDC service before it is ready:
`VARY WLM,APPLENV=<applic.env-name>,Q`
Repeat this for all application environments using the JCL procedure(s) which specify the LOAD library of the version being replaced. If the host is a Parallel Sysplex, precede the command with `ROUT *ALL`, routing it to all systems in plex.
2. Use the `R4ZUTIL TERMINATE` verb to free all system resources currently used by the CDC service. Repeat this in all LPARs where it may be executing (if the source DB2 has data sharing members in those systems).

Controlling the CDC process

Replicate uses ECSA memory structures - called R4Z CDC services - which are formed during CDC processing. Each z/OS LPAR may contain several such R4Z CDC services. To form the R4Z CDC services, a special utility program, R4ZCTL, is provided.

This program can also be used for managing the state of CDC processing, as well as the level of traffic on the z/OS side.

The R4Z control program (R4ZCTL) is required for the following purposes: checking CDC and reporting the CDC service's status, pausing CDC processing, resuming a paused CDC, and terminating the CDC process, either normally or forcefully.

The command parameters available when running the R4ZCTL program are:



If you run the command without any parameters, it will return the current status of the CDC service.

- `CHECKCONFIG` - Verifies that configuration parameters are valid
- `PAUSE_TASK` - Stops the service of a specific CDC task
- `PAUSE_CDC` - Stops the entire activity of a CDC service
- `RESUME_TASK` - Enables a paused task to continue
- `RESUME_CDC` - Enables a paused CDC service to continue
- `TERMINATE` - Stops the CDC service, and, once stopped – releases its resources
- `FORCE` - same as `TERMINATE`, when task holding resources “hangs”

As the R4ZCTL programs requires APF-authorization, all libraries in the STEPLIB must be APF- authorized.

When running R4ZCTL with no parameter, its completion-code is set based on whether the environment was initialized (`CC=0`) or not initialized (`CC=1`).

Control program invocation syntax

The R4ZCTL program is invoked as a job step, i.e. `EXEC PGM=R4ZCTL`, and accepts instructions via the invocation parameter of the job step.

9 Adding and managing source endpoints

The R4ZCTL invocation parameter is a string that can contain optional sub-parameters, separated by a comma as follows:

```
[SERVICE=CDC-service-qualifier,][MSGLVL={0 | 1},][action]
```



The order of the sub-parameters in the string is not important.

Where:

`SERVICE=CDC-service-qualifier` designates the logical scope of Replicate activity, upon which the control program is to act.

`MSGLVL={0 | 1 | 2}` designates the level of notifications to be displayed in the message file during the operation of the control program.

- 0 = No notifications
- 1 = Moderate
- 2 = Maximum

`action-verb` can be one of the following:

- `PAUSE_TASK(*|ALL|task-qualifier)`
Suspends CDC retrieval for the task(s) designated in parentheses.
- `PAUSE_CDC`
Suspends CDC retrieval for entire CDC service. Replication tasks suspended for more than a certain time are stopped will attempt recovery multiple times. No new tasks will be served.
- `RESUME_TASK(*|ALL|task-qualifier)`
Resumes CDC retrieval for the task designated by `task-qualifier`.
- `RESUME_CDC`
Resumes CDC retrieval for all instances matching the `session-limits` qualifier.
- `DUMP_TASK(*|ALL|task-qualifier)`
Requests formatted dumping of the control information in the resident memory structures. `ALL` designates all sessions; `SUMMARYONLY` designates only the anchor.
- `TERMINATE`
Frees all the R4Z resident memory structures, terminates all active instances and deletes associated resources. From this point on, all CDC requests will return the inactive status until CDC service is auto-activated by an incoming call to the CDC reader UDF.

Syntax elements reference

The elements used in the syntax descriptions above are as follows:

- `CDC-service-qualifier`
A 4-character name (first character alphabetic, rest alphanumeric), which designates a CDC service. "CDC service" refers to a group of resources – memory structures, a user-defined table-function (UDTF) defined in DB2 and an application environment (APPLENV) defined in the WLM policy. R4Z lets you specify properties – memory limits and processing thresholds – per each CDC service. It may be

necessary to manage several CDC services for another reason as well; namely, to allow for more than one Replicate release being used in a single LPAR, with each release maintaining its memory structures independently of others.

- **task-qualifier**

A 4-digit identifier being assigned upon initiation of the task. This identifier is aimed mainly for controlling and tracking CDC traffic of a specific task. It forms the CORRID (Correlation-ID) of the DB2 thread serving CDC – it occupies bytes 9-12 there.

Control program completion codes

- 0 – Normal completion
- 1 – Environment does not exit the `session-limits` qualifier (when no action is specified)
- 4 – Warning
- 8 – Error

Sample jobs (in the INSTALL library)

IV1CHECK – Checks the configuration of all CDC services

XMDUMP – Dumps all sessions

XMPAUSE – Pauses a session

XMRESUME – Resumes a session

XMTERMIN – Terminates all sessions

Enabling the CDC process (auto-activation)

To enable the CDC reader UDTF function to work, this function needs to be able to allocate and pre-format memory structures in ECSA when the UDTF is called and the first time after the z/OS system was started; it also needs to access the configurations set for the CDC service the UDTF is serving (or the “default” configurations, for values not specified at the CDC service level).

To account for this, the installation process ensures that:

1. A special UDTF is created to serve the CDC service. The created UDTF has its name suffixed with a double-underscore (“__”) followed by the `CDC-service-qualifier`, forming a name as such:
`<schema>.R4Z_UDTF__<CDC-service-qualifier>`.
2. A WLM APPLENV is created to execute invocations of the above UDTF. The created APPLENV may also have its name suffixed with the CDC service qualifier, to ensure uniqueness of the APPLENV name – which is recommended; however, multiple UDTFs may use a single APPLENV.
3. A JCL procedure is created to “host” the WLM APPLENV executions. This JCL procedure differs from a usual WLM STC procedures in that a special DDNAME, `R4ZCNFG`, must be specified, its DSN referring to the CONFIG library of the R4Z product. In this library, there MUST be an existing member named “CDCS”, specifying the default CDC service configurations. There MAY also exist a member named “CDCS<CDC-service-qualifier>” with the configuration values you want to apply for this CDC service.
4. The content of each of the CDCS* members in the CONFIG library is a list of assignment statements, one per card, in the format “keyword=value”. The statement may follow spaces, and are space-terminated; no spaces are allowed with the statement. Cards beginning with a hyphen (‘-’) are treated

9 Adding and managing source endpoints

as comment cards; and the content following the terminating space is also treated as comment.

Keywords, max values, min values, and default values

Keyword	Description	Minimum	Maximum	Default
MAXSESSIONS	The maximum number of CDC sessions.	1	128	32
SESSIONTIMEOUTSECS	The number of seconds after which session is timed out.	300	7200	900
MAXIFIBUFKBYTES	The maximum buffer size allowed for IFI reads, in Kbytes.	64	1024	256
MAXRESULTSETMBYTES	The maximum size in Mbytes for the accumulated result set to reach; when reached, the result set is returned.	1	2048	4
MAXRESULTSETSECS	The maximum time interval, in seconds, for accumulating the result set; when reached, the result set is returned.	1	120	30

Establishing R4Z CDC services

Deciding on the CDC services: Based on predicting the need for independent replication processes (to allow different versions, to ensure sufficient ECSA memory, to account for different latencies allowed, etc.), determine which R4Z CDC services are to exist in each LPAR. For each service, choose the CDC service qualifiers (4 characters). You also need to decide whether each CDC service is to have its own WLM APPL ENV (application environment), or a single APPL ENV to serve all CDC services. You need to set variable DEFWLMAE (DEFine WLM Applic. Env.) in DFSYMLST member to either PER (first option) or ONE (second option). Once set, you can proceed with the installation:



- If you select ONE for DEFWLMAE, jobs DO2* and DO3SRVTF should be run one time ; if you select PER , these jobs should be run for each CDC service, after the JCL SET variable CDCSRV is set to the CDC service's qualifier.
- Job DO3SRVDF creates a CONFIG library, and places a member named "CDCS", which serves as the default configuration file. After running job DO3SRVTF the library will contain the configuration of all CDC services, making it possible to edit them and change the configurations of a specific CDC service.
- If ONE is used for DEFWLMAE, then the product defaults are used instead of the CONFIG library members.

9.14 Using IBM Informix as a source

This section describes how to set up and use an IBM Informix database as the source database in a replication task.

In this section:

- *Prerequisites (page 307)*
- *Limitations and considerations (page 307)*
- *Required permissions (page 308)*
- *Supported data types (page 308)*
- *Setting general connection properties (page 309)*
- *Setting advanced connection properties (page 310)*

Prerequisites

Before you begin to work with an IBM Informix database as a source in Qlik Replicate, make sure that the following prerequisites have been met:

- **Qlik Replicate machine:**
 - Qlik Replicate installed in your network.
 - IBM Informix ODBC Driver (64-bit) version 3.70 or later must be installed on the computer where Qlik Replicate is located.
 - The `DB_LOCALE=<Informix_db_locale_value>` environment variable must be set to be the name of the IBM Informix database locale from which you want to capture data.
 - The `INFORMIXSQLHOSTS` environment variable must be set to include the names of the IBM Informix servers that are available for use as Replicate source endpoints.
- **IBM Informix Server:**
 - An IBM Informix account with the required *Required permissions (page 308)*.
 - CDC enabled. To enable CDC, run the script `$INFORMIXDIR/etc/syscdcv1.sql` on the IBM Informix server.



This requires DBA privileges (User 'IBM Informix' or another DBA user).

- Make sure that the database to be replicated was created with either the `WITH LOG` or the `WITH BUFFERED LOG` property.

Limitations and considerations

When using IBM Informix as a database in a Replicate task, the following limitations currently apply:

- CDC does not capture DDL changes. Due to an IBM Informix limitation, IBM Informix does not allow DDLs to be executed on tables with Full Row Logging enabled. To learn how to capture DDL changes during CDC, see [Automatically enable full row logging](#).
- Due to an IBM Informix limitation, columns that follow columns of data types `SET`, `MULTISET` or `LIST` will not be replicated during CDC. For example, in the table below, changes to Col3 will not be captured during CDC.

Example table

Name	Data Type
Col1	INTEGER
Col2	SET
Col3	INTEGER

- User-defined data types are not supported.
- Start from timestamp is not supported.



Choosing this option will resume the task from the current time.

9 Adding and managing source endpoints

- If a task with an IBM Informix source is stopped before any changes have been made and then resumed, any changes that were made between the time that the task was stopped and the time that it was resumed will be lost.
- Due to a known issue with the IBM Informix CDC API, Replicate does not support replication of tables whose names contain spaces or non-English letters.
- All transactions should be manually stopped before starting a Full Load task. Transactions started during the Full Load operation will be cached.
- FIELDPROC and EDITPROC are not supported during CDC.
- Replication from an IBM Informix High Availability cluster is not supported.
- Replicating data with 4-byte emoji characters is not supported.

Required permissions

In order to access the specified database, the user specified in the **General** tab must be a member of the "IBM Informix" group (which has DBA privileges) on the database server.

Supported data types

The following table shows the IBM Informix database source data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.

For information on how to view the data type that is mapped in the target, see the section for the target database you are using.

For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).

Supported data types

IBM Informix Source Data Types	Qlik Replicate Data Types
INTEGER	INT4
SMALLINT	INT2
INT8	INT8
SERIAL	INT4
SERIAL8	INT8
NUMERIC (p,s)	NUMERIC (p,s)
DECIMAL (p,s)	NUMERIC (p,s)
MONEY (p,s)	NUMERIC (p,s)
FLOAT	REAL8
DOUBLE	REAL8
REAL	REAL4
SMALLFLOAT	REAL4
BIGINT	STRING (20)

IBM Informix Source Data Types	Qlik Replicate Data Types
DATE	DATE
DATETIME (fraction)	DATETIME (fraction)
INTERVAL	STRING
CHAR	STRING (n)
VARCHAR (n)	STRING (n)
LVARCHAR (n)	STRING (n)
NCHAR (n)	STRING (n)
NVARCHAR (n)	STRING (n)
BLOB	BLOB
BYTE	BLOB
CLOB	CLOB
LIST <i>See also Limitations and considerations (page 307).</i>	CLOB
MULTISET <i>See also Limitations and considerations (page 307).</i>	CLOB
SET <i>See also Limitations and considerations (page 307).</i>	CLOB
TEXT	CLOB
BOOLEAN	BOOLEAN

Unsupported data types

The following IBM Informix data types are not supported:

- Any user-defined data type

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.



You can also use IBM Informix files as a source. For more information, see Using the Qlik Replicate file channel (page 1061).

To add an IBM Informix source endpoint to Qlik Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the IBM Informix database. This is optional.
4. Select **Source** as the database **role**.
5. Select **IBM Informix** as the database **Type**.



*When this endpoint is used as a duplicated source in a Log Stream Staging setup, select the **Read changes from log stream** check box and then select the relevant Log Stream Staging task from the drop-down list.
For information on setting up and managing Log Stream Staging tasks, see *Using the Log Stream* (page 1072).*

6. In the **Server** field, enter the name of the IBM Informix server. On Windows, this must correspond to one of the hosts defined using the `setnet32.exe` tool. On Linux, this must correspond to a valid `dbservername` entry in the `$INFORMIXDIR/etc/sqlhosts` file on the computer running the application.



Consider the following:

- *This information is case sensitive.*
- *To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**.
If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.
To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.*

7. Enter the IBM Informix authentication information (**User Name**, **Password**) for the authorized user for this IBM Informix database. If you do not know this information, see your IBM Informix database administrator (DBA).



*Make sure that the IBM Informix user entered in the IBM Informix Authentication section has the correct access privileges. For information on how to provide the required privileges, see *Required permissions* (page 308).*

8. In the **Database name** field, enter the IBM Informix database name.

Setting advanced connection properties

In the **Advanced** tab, you can set the following parameters:

- **Automatically enable full row logging:** Full Row Logging is required for CDC. Select this option to automatically enable Full Row Logging for the tables to be replicated. To automatically enable Full Row Logging, the user specified in the **General** tab must have administrative privileges on the IBM Informix database.



DDL events are not captured during CDC. To perform DDL operations on source tables in a Replicate CDC task:

1. *Stop the Replicate task.*
2. *Disable Full Row Logging for the relevant tables as in the following example:
execute function syscdcvt1:IBM Informix.cdc_set_fullrowlogging
('sysuser:IBM Informix.employees_table', 0)*
3. *Perform the DDL operation(s).*
4. *If the **Automatically enable full row logging** option is not selected, manually enable Full Row Logging for the relevant tables.*
5. *Start the Replicate task.*
6. *Reload the relevant tables or perform a Full Load.*

- **Max bytes per read:** Specify the maximum number of bytes to read each time the log is accessed. If you encounter performance issues, adjusting this number may help.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

9.15 Using Microsoft Azure SQL Managed Instance as a source

This section describes how to set up and use a Microsoft Azure SQL Managed Instance database as the source database in a replication task.



While it is certainly possible to capture changes from a Microsoft Azure SQL Managed Instance, due to its [limitations](#), it is recommended to use the newer Microsoft Azure SQL (MS-CDC) endpoint instead.

In this section:

- *Prerequisites (page 313)*
- *Limitations and considerations (page 316)*
- *Required permissions (page 317)*
- *Supported compression methods (page 317)*
- *Microsoft Azure SQL Managed Instance source data types (page 318)*
- *Non-supported data types (page 320)*
- *Homogeneous replication (page 321)*
- *Setting general connection properties (page 322)*
- *Setting advanced connection properties (page 323)*
- *Artifacts created by Replicate in the database (page 324)*

Prerequisites

Make sure that the following prerequisites have been met:

Qlik Replicate for Windows

Install Microsoft ODBC Driver 18.1 for Windows on the Qlik Replicate Server machine.

Qlik Replicate for Linux

The procedure below assumes that you have installed a single default instance of Replicate on Linux (areplicate). If you have [installed multiple instances](#), replace areplicate with the name of the instance running the task with a Microsoft SQL Server source. If several instances are running such as task, the procedure needs to be repeated for each instance.

Install Microsoft ODBC Driver 18.1 for Linux on the Replicate machine.

Then, open a Unix shell and do the following:

1. Change the working directory to `<Replicate-Install-Dir>/bin`.
2. Stop the Replicate service:

```
./areplicate stop
```
3. Optionally, confirm that the service has stopped:

```
./areplicate status
```
4. Copy the driver location to the `site_arep_login.sh` file as follows:

```
echo "export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/microsoft/msodbcsql18/lib64/" >>  
site_arep_login.sh
```

This will add the driver to "LD_LIBRARY_PATH" and update the driver location in the **site_arep_login.sh** file.
5. Optionally, confirm that the driver location was copied:

```
cat site_arep_login.sh
```
6. Start the Replicate service:

```
./areplicate start
```
7. Optionally confirm that the service has started:

```
./areplicate status
```
8. Replicate requires the following ODBC library: `libmsodbcsql-18.1.so.1.1`
To check which library version is currently installed Issue the following command:

```
ls /opt/microsoft/msodbcsql/lib64/
```

If the existing library has a different version number (e.g. `libmsodbcsql-18.0.so.1.1`), you need to create a symbolic link between the existing library and the required library.
To do this, issue the following commands:

```
cd /opt/microsoft/msodbcsql18/lib64/  
ln -s existing_library_name libmsodbcsql-18.1.so.1.1
```

where *existing_library_name* is the name of the currently installed library (e.g. `libmsodbcsql-18.0.so.1.1`).

Other prerequisites

- A Microsoft Azure SQL Managed Instance account with the specific access privileges is required. See [Source Permissions](#) for more information.

- When Replicate is installed on Linux, the Replicate machine and the source database must both be part of the Azure domain.
- Microsoft Azure SQL Managed Instance as a source must be configured for a full backup to work with Qlik Replicate. For more information, see *Preparing Microsoft Azure SQL Managed Instance backup and recovery* (page 314).

Preparing the Microsoft Azure SQL Managed Instance database for replication

This topic describes the configuration requirements for using a Microsoft Azure SQL Managed Instance database. A Microsoft Azure SQL Managed Instance system administrator should carry out these tasks.

Setting up a Microsoft Azure SQL Managed Instance for CDC

The following section explains how to set up a Microsoft Azure SQL Managed Instance for CDC.



While it is certainly possible to capture changes from a Microsoft Azure SQL Managed Instance, due to its limitations (see below), it is recommended to use the newer Microsoft Azure SQL (MS-CDC) endpoint instead.



When creating the Microsoft Azure SQL Managed Instance, full backup must be enabled; otherwise, Replicate will not be able to capture changes.

To set up a Microsoft Azure SQL Managed Instance for CDC:

1. Log in as the master user in the DB instance and set up the database for MS-CDC as follows:

```
exec source_db_name.dbo.sp_cdc_enable_db
```

where *source_db_name* is the name of the source database.
2. To enable MS-CDC, run the following for each of the intended source tables:

```
exec sys.sp_cdc_enable_table  
@source_schema = N'schema_name',  
@source_name = N'table_name',  
@role_name = NULL  
exec sys.sp_cdc_add_job @job_type = 'capture'  
exec sys.sp_cdc_add_job @job_type = 'cleanup'  
GO
```

where *schema_name* and *table_name* are the names of the source schema and the source table respectively.
3. Set the retention period for the changes to be available on the source by changing the `@pollinginterval` of the 'capture' job to the maximum possible value.

Limitations

- CDC is supported with the transaction log only, with no option to work with backup.
- Events may be lost if they are moved to backup or truncated.

Preparing Microsoft Azure SQL Managed Instance backup and recovery

Qlik Replicate consumes changes captured from the database transaction log (TLOG). The TLOG is maintained by Microsoft Azure SQL Managed Instance for recovery purposes. All changes made to a database are written to the TLOG. The following happens when recovery is required:

- A backup copy of the database is made.
- Logged events are taken and used in a rolling-forward process where the recorded changes are replayed against that copy.

To prepare for backup and recovery you must make sure that the Microsoft Azure SQL Managed Instance Recovery Model is set up. You select the Recovery Model in the Microsoft Azure SQL Managed Instance Management Studio. This should be carried out by a Microsoft Azure SQL Managed Instance system administrator.

The TLOG data is truncated as soon as it is no longer needed therefore the TLOG is not persistent. However, Qlik Replicate guaranteed delivery requires persistency in the changed data. To ensure persistency:

- A full database backup must be carried out before beginning to replicate data.
- The Recovery Model must be set to **Bulk logged** or **Full**.

To set the recovery model

In the database properties **Options** tab, set the Recovery Model to **Bulk logged** or **Full**. In these modes, the transaction Log is more durable.



After setting the Recovery Model, it is strongly recommended not to change it; doing so may result in loss of data.

Defining Microsoft Azure SQL Managed Instance database settings

Set the following for the Microsoft Azure SQL Managed Instance database(s) that you are using as a source:

- From the Object Explorer in the Microsoft Azure SQL Managed Instance Management Studio, right click the database and select **Properties**. In the **Options** tab, set the **Recovery model** to **Bulk logged** or **Full**. In this mode, the transaction Log is more durable and truncation occurs less frequently.
- Ensure that there is a full database backup for each Microsoft Azure SQL Managed Instance database that you are using as a source.
- When creating a connection string, it is possible to use any parameter supported by Microsoft Azure SQL Managed Instance. The Microsoft Azure SQL Managed Instance system administrator must ensure that the Microsoft Azure SQL Managed Instance instance is configured correctly so that the proper authentication credentials are accepted.

Working with Windows Authentication

You can configure the Qlik Replicate Microsoft Azure SQL Managed Instance endpoint to log in to Microsoft Azure SQL Managed Instance (on Windows) using Windows authentication.

If you choose this option, you also need to make sure that:

- The Microsoft Azure SQL Managed Instance instance is set up to allow Windows log on.
 - The Qlik Replicate user is specified as the "Log on as" user for the "Qlik Replicate Server" service account.
- OR-

Microsoft Azure SQL Managed Instance is configured to allow login for the Qlik Replicate Server service account.

Limitations and considerations

When using a Microsoft Azure SQL Managed Instance source endpoint in a Replicate task, the following limitations apply:

- A Secondary SQL Server database is not supported as a source database.
- If you are using a Microsoft Azure SQL Managed Instance source database in a replication task, the Microsoft Azure SQL Managed Instance Replication Publisher definitions for the database that was used in the task are not removed when you remove a task. A Microsoft Azure SQL Managed Instance system administrator must delete these definitions from Microsoft Azure SQL Managed Instance.
- Sparse tables are not supported.
- Replicating data from indexed views is not supported.
- Renaming tables using `sp_rename` is not supported (e.g. `sp_rename 'Sales.SalesRegion', 'SalesReg';`)
- Renaming columns using `sp_rename` is not supported (e.g. `sp_rename 'Sales.Sales.Region', 'RegID', 'COLUMN';`)
- TRUNCATE events will not be captured.
- Replication of computed columns is supported in Full Load replication tasks only. During CDC, INSERTs to computed columns will appear as NULL on the target and UPDATEs will be ignored.
- Microsoft Azure SQL Managed Instance partition switching is not supported.
- When using the WRITETEXT and UPDATETEXT utilities, Qlik Replicate does not capture events applied on the source database.
- The following DML pattern is not supported:
`select <*> into <new_table> from <existing_table>`
- Column-level encryption is not supported.
- The following limitations apply when accessing the backup transaction logs:
 - Encrypted backups are not supported.
 - Backups stored at a URL or on Windows Azure are not supported.
- Microsoft Azure SQL Managed Instance backup to multiple disks is not supported.
- When inserting a value into SQL Server spatial data types (GEOGRAPHY and GEOMETRY), one can either ignore the SRID (Spatial Reference System Identifier) property - in which case the default SRID will be used (0 for GEOMETRY and 4326 for GEOGRAPHY) - or specify a different number. When replicating tables with spatial data types, Qlik Replicate replaces the SRID that was inserted by user with the default SRID.
- Replicating GEOGRAPHY and GEOMETRY data types is only supported in limited LOB mode.
- Columnstore indexes are not supported.
- Memory-optimized tables (using In-Memory OLTP) are not supported.
- Temporal databases are not supported
- Delayed durability is not supported
- Table change tracking is not supported

- Masked data will be created in the target without masking.
- Replication from databases configured to use ADR (Accelerated Database Recovery) is not supported.
- Capturing DDL and DML statements within a single transaction is not supported.
- Replicating a table or schema with an apostrophe (') in its name is not supported.
- UPDATES to a Primary Key/Unique Index that affect multiple rows may cause conflicts when applying the changes to the target. This may occur, for example, if the UPDATES are applied to the target as INSERT/DELETE operations rather than a single UPDATE operation. In such a scenario (where UPDATES to a Primary Key/Unique Index affect multiple rows), working in Batch optimized apply Change Processing mode is not supported as it may result in the table being ignored. Working in Transactional Apply Change Processing mode however, may result in constraint violations. If this happens, you either need to reload the relevant table or locate the problematic records in the *Apply exceptions* (page 1298) Control Table and edit them manually in the target database. See also: *Change Processing Tuning* (page 1181).
- Changes applied to the database using DACPAC will not be captured.
- System tables and views are not supported.

Non-supported Microsoft Azure SQL Managed Instance security features

Tables that use the following Microsoft Azure SQL Managed Instance security features are not supported:

- Always Encrypted
- Row-Level Security

Required permissions

To use a Microsoft Azure SQL Managed Instance source in a Qlik Replicate task, the user specified in the Microsoft Azure SQL Managed Instance [endpoint connection settings](#) must be a member of the db_owner database role.

To use the db_owner database role, the following permissions are required on the master DB:

- `grant view any definition`
- `grant view server state`
- `grant execute on sp_repldone`
- `grant execute on sp_replincrementlsn`
- `grant execute on sp_addpublication`
- `grant execute on sp_addarticle`
- `grant execute on sp_articlefilter`
- `grant select on sys.fn_dblog`

Supported compression methods

Supported compression methods

Row/Page Compression (at Partition Level)	Vardecimal Storage Format	Vardecimal Storage Format Sparse Columns	Vardecimal Storage Format Sparse Columnar Structure Compression
Yes	No	No	No

Microsoft Azure SQL Managed Instance source data types

The Microsoft Azure SQL Managed Instance source for Qlik Replicate supports most Microsoft Azure SQL Managed Instance data types. The following table shows the Microsoft Azure SQL Managed Instance source data types that are supported when using Qlik Replicate and the default mapping to Qlik Replicate data types. Note that Microsoft Azure SQL Managed Instance data types are only mapped to Qlik Replicate data types when the target endpoint is not Microsoft Azure SQL Managed Instance. For information on data type mapping and collation support when the target endpoint is Microsoft Azure SQL Managed Instance, see *Homogeneous replication (page 321)*.

For information on how to view the data type that is mapped in the target, see the section for the target endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.



When row size exceeds 8060, the VARCHAR, NVARCHAR, and VARBINARY data types will be stored out-of-row. To replicate such data, use the source lookup function described in Data Enrichment (page 1157).



Collatable data types are indicated by an asterisk ().*

Supported data types

Microsoft Azure SQL Managed Instance Data Types	Qlik Replicate Data Types
BIGINT	INT8
BIT	BOOLEAN
DECIMAL	NUMERIC
INT	INT4
MONEY	NUMERIC (19,4)
NUMERIC (p,s)	NUMERIC
SMALLINT	INT2
SMALLMONEY	NUMERIC (10,4)
TINYINT	UINT1
REAL	REAL4
FLOAT	REAL8

Microsoft Azure SQL Managed Instance Data Types	Qlik Replicate Data Types
DOUBLE	REAL8
DATETIME	DATETIME
DATETIME2	DATETIME
SMALLDATETIME	DATETIME
DATE	DATE
TIME	STRING (16)
DATETIMEOFFSET	STRING
*CHAR	STRING
*VARCHAR	STRING
*VARCHAR (max) *TEXT To use this data type with Qlik Replicate, you must enable the use of CLOBs for a specific task. LOB columns for Microsoft Azure SQL Managed Instance tables are updated in the target even for UPDATE statements that did not change the value of the LOB column in Microsoft Azure SQL Managed Instance. During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.	CLOB
*NCHAR	WSTRING
*NVARCHAR (length)	WSTRING
*NVARCHAR (max) *NTEXT To use this data type with Qlik Replicate, you must enable the use of NCLOBs for a specific task. LOB columns for Microsoft Azure SQL Managed Instance tables are updated in the target even for UPDATE statements that did not change the value of the LOB column in Microsoft Azure SQL Managed Instance. During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.	NCLOB
BINARY	BYTES

Microsoft Azure SQL Managed Instance Data Types	Qlik Replicate Data Types
VARBINARY	BYTES
VARBINARY (max) IMAGE <p>LOB columns for Microsoft Azure SQL Managed Instance tables are updated in the target even for UPDATE statements that did not change the value of the LOB column in Microsoft Azure SQL Managed Instance.</p> <p>To use this data type with Qlik Replicate, you must enable the use of BLOBs for a specific task.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p>	BLOB
TIMESTAMP	BYTES
UNIQUEIDENTIFIER	STRING
HIERARCHYID	VARCHAR (x)
XML <p>LOB columns for Microsoft Azure SQL Managed Instance tables are updated in the target even for UPDATE statements that did not change the value of the LOB column in Microsoft Azure SQL Managed Instance.</p> <p>To use this data type with Qlik Replicate, you must enable the use of NCLOBs for a specific task.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p>	CLOB
GEOMETRY	CLOB
GEOGRAPHY	CLOB

Non-supported data types

Tables that include fields with the following data types are not supported by Qlik Replicate.

- CURSOR
- SQL_VARIANT
- TABLE



User-defined data types are supported according to their base-type. For example, a user-defined data type based on DATETIME is handled as a DATETIME data type.

Homogeneous replication

The following section describes how Replicate handles replication from Microsoft Azure SQL Managed Instance to the following targets: Microsoft SQL Server, Microsoft Azure SQL Database, Microsoft Azure SQL Managed Instance, and Google Cloud SQL for SQL Server.

In homogeneous replication, source column and table collations will be replicated to the target as described in *Column and table collation* (page 322).



In homogeneous replication, the source data first passes through the Qlik Replicate data type and is therefore subject to any limitations of that type.

For information on Replicate data types and their limitations (where relevant), see *Replicate data types* (page 81).

For information on which Replicate data types the source data passes through when replicating from a Microsoft Azure SQL Managed Instance, see the *Microsoft Azure SQL Managed Instance to Qlik Replicate data types mapping table* described earlier.



To prevent data truncation when replicating columns with an **XML** data type, it is strongly recommended to enable the [Allow unlimited LOB size](#) option in the task settings.

Data type exceptions

When replicating from Microsoft Azure SQL Managed Instance to an SQL Server-based target, source and target data types are identical, with the following exceptions:

Data type exceptions in homogeneous replication

Microsoft Azure SQL Managed Instance Source	Microsoft SQL Server-based Target
VARCHAR	VARCHAR (x) (when x=0 or x>8000) else VARCHAR (max)
NVARCHAR (length)	NVARCHAR (x) (when x=0 or x>8000) else NVARCHAR (max)

Microsoft Azure SQL Managed Instance Source	Microsoft SQL Server-based Target
VARBINARY	VARBINARY (x) (when x=0 or x>8000) else VARBINARY (max)
HIERARCHYID	VARCHAR (x)
TIMESTAMP	VARBINARY

Column and table collation

When replicating from Microsoft Azure SQL Managed Instance to an SQL Server-based target, column and table collations will be replicated to the target.



To support collation replication, the DBA must ensure that the collations defined for the source Microsoft Azure SQL Managed Instance database are the same as those defined for the SQL Server-based target.

Non-nullable columns and Primary/Unique Index names

Primary/Unique Index names are preserved during homogeneous replication. Non-nullable columns are also preserved during homogeneous replication, with the exception of the following data types:

- text
- ntext1
- varchar(max)
- nvarchar(max)
- varbinary(max)
- image
- xml

Setting general connection properties

This section describes how to configure connection properties.

To add a Microsoft Azure SQL Managed Instance source endpoint to Qlik Replicate:

1. In the Qlik Replicate Console, click Manage Endpoint Connections to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Microsoft Azure SQL Managed Instance. This is optional.
4. Select **Source** as the database **role**.
5. Select **Microsoft Azure SQL Managed Instance** as the database **Type**.

9 Adding and managing source endpoints

- Specify the **Server name**. This is the host name or IP address of the computer with the Microsoft Azure SQL Managed Instance.



- To override the default port, add the port to the server name, separated by a comma. For example, if the server name is `myserver.company.local` and the port is `3333`, then the server name should be entered like this:
`myserver.company.local,3333`

- Select one of the following authentication methods:
 - SQL Authentication
 - Azure Active Directory Authentication
 - Azure Active Directory Service Principal
- In the **Username** field, enter your user name for accessing the database. When **Azure Active Directory Service Principal** is selected, this is the Client ID.
- In the **Password** field, enter your password for accessing the database. When **Azure Active Directory Service Principal** is selected, this is the Secret Key.



To ensure that the password provided to Replicate is never used by any other principal, Replicate can be configured to automatically replace the user-entered password with a strong random password. For more information, see [Configuring Replicate to Automatically Replace the User-Entered Password](#).



- The username and password are case sensitive.
- To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**. If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box. To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

- Type the **Database name** or click **Browse** and select one from the list of available databases. This is the name of the database from where you are replicating the data.

Setting advanced connection properties

In the **Advanced** tab, you can set internal parameters, set more options, and view a summary of your settings.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

Artifacts created by Replicate in the database

In order to capture changes, Qlik Replicate creates various artifacts in the **dbo** schema of the Microsoft Azure SQL Managed Instance when the task starts. You can safely remove them when the task completes.

The following artifact is created to track open transactions:

- #attrep_dbcc_open_trans_table

The following functions are created to verify that the LSN (Log Sequence Number) is valid:

- attrep_fn_LsnSegmentToHexa
- attrep_fn_NumericLSnToHexa

9.16 Using Microsoft Azure SQL (MS-CDC) as a source

This section describes how to set up Microsoft Azure SQL (MS-CDC) as a source endpoint in a replication task.

When replicating data from Microsoft Azure SQL Managed Instance or Microsoft Azure SQL Database (using Microsoft Azure SQL (MS-CDC)), changes are captured from Microsoft Change Tables as opposed to the SQL Server transaction log. Eliminating the need to rely on the transaction log minimizes the risk of data loss resulting from online transaction log retention time and backup transaction log inaccessibility.



Replication from a geo-replica database is also supported. For more information about geo-replication, see <https://learn.microsoft.com/en-us/azure/azure-sql/database/active-geo-replication-configure-portal?view=azuresql&tabs=portal>

In this section:

- *Prerequisites (page 326)*
- *Limitations and considerations (page 327)*
- *Required permissions (page 329)*
- *Supported compression methods (page 329)*
- *Supported data types (page 330)*
- *Homogeneous replication (page 333)*
- *Setting general connection properties (page 334)*
- *Setting advanced connection properties (page 346)*
- *Removing Replicate artifacts from the database (page 347)*

Prerequisites

The following section describes the prerequisites for working with the Qlik Replicate Microsoft Azure SQL (MS-CDC) endpoint.

Client Prerequisites

Qlik Replicate for Windows

Install Microsoft ODBC Driver 18.1 for Windows on the Qlik Replicate Server machine.

Qlik Replicate for Linux

The procedure below assumes that you have installed a single default instance of Replicate on Linux (areplicate). If you have [installed multiple instances](#), replace areplicate with the name of the instance running the task with a Microsoft SQL Server source. If several instances are running such as task, the procedure needs to be repeated for each instance.

Install Microsoft ODBC Driver 18.1 for Linux on the Replicate machine.

Then, open a Unix shell and do the following:

1. Change the working directory to `<Replicate-Install-Dir>/bin`.
2. Stop the Replicate service:
`./areplicate stop`
3. Optionally, confirm that the service has stopped:
`./areplicate status`
4. Copy the driver location to the `site_arep_login.sh` file as follows:
`echo "export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/microsoft/msodbcsql18/lib64/" >> site_arep_login.sh`
This will add the driver to "LD_LIBRARY_PATH" and update the driver location in the **site_arep_login.sh** file.
5. Optionally, confirm that the driver location was copied:
`cat site_arep_login.sh`
6. Start the Replicate service:
`./areplicate start`
7. Optionally confirm that the service has started:
`./areplicate status`
8. Replicate requires the following ODBC library: `libmsodbcsql-18.1.so.1.1`
To check which library version is currently installed Issue the following command:
`ls /opt/microsoft/msodbcsql/lib64/`
If the existing library has a different version number (e.g. `libmsodbcsql-18.0.so.1.1`), you need to create a symbolic link between the existing library and the required library.
To do this, issue the following commands:
`cd /opt/microsoft/msodbcsql18/lib64/`
`ln -s existing_library_name libmsodbcsql-18.1.so.1.1`
where `existing_library_name` is the name of the currently installed library (e.g. `libmsodbcsql-18.0.so.1.1`).

Preparing the database for CDC

The following section explains how to set up the database for CDC.

To set up the database for CDC:

1. Set up the database for MS-CDC by executing the following:

```
exec source_db_name.dbo.sp_cdc_enable_db
```

where `source_db_name` is the name of the source database.

2. The replication task automatically enables MS-CDC for all the source tables. However, if you prefer to enable MS-CDC manually, execute the following script (for each of the intended source tables):

```
exec sys.sp_cdc_enable_table
```

```
@source_schema = N'schema_name',
```

```
@source_name = N'table_name',
```

```
@role_name = NULL
```

where `schema_name` and `table_name` are the names of the source schema and the source table respectively.



- The CT table names must use the default naming format: `<SchemaName>_<TableName>_CT`
- The CT table must include the following columns: `[__$start_lsn]`, `[__$end_lsn]`, `[__$seqval]`, `[__$operation]`, `[__$update_mask]`, `<all the data columns>`, `[__$command_id]`

3. Make sure the MS-CDC capture and cleanup jobs are running.



- You can control the jobs using `sys.sp_cdc_change_job`. For additional information, see the Microsoft Help.
- To reduce latency, you might want to set the `@pollinginterval` parameter to a small value.

TLS 1.2 prerequisites

If your environment matches the following:

- Microsoft SQL Server 2014
- Microsoft ODBC Driver 18 installed on Red Hat 8.1 or later

Then, in order to work with TLS 1.2, you must install a specific Service Pack on the Windows machine where Microsoft SQL Server is installed.

For more information, see <https://support.microsoft.com/en-us/help/3135244/tls-1-2-support-for-microsoft-sql-server>.

Limitations and considerations

You should be aware of the following limitations and considerations when using a Microsoft Azure SQL (MS-CDC) source:

- A Secondary Microsoft Azure SQL Managed Instance is not supported as a source database.
- Sparse tables are not supported.
- Replicating data from indexed views is not supported.
- Replication of computed columns is supported in Full Load replication tasks only. During CDC, INSERTs to computed columns will appear as NULL on the target and UPDATES will be ignored.
- Partition switching is not supported.
- The following DML pattern is not supported:

```
select <*> into <new_table> from <existing_table>
```
- Column-level encryption is not supported.
- When inserting a value into Microsoft Azure SQL Managed Instance spatial data types (GEOGRAPHY and GEOMETRY), one can either ignore the SRID (Spatial Reference System Identifier) property - in which case the default SRID will be used (0 for GEOMETRY and 4326 for GEOGRAPHY) - or specify a different number. When replicating tables with spatial data types, Replicate replaces the SRID that was inserted by user with the default SRID.
- Columnstore indexes are not supported.
- Memory-optimized tables (using In-Memory OLTP) are not supported.
- Temporal databases are not supported
- Delayed durability is not supported
- Due to an ODBC limitation, no more than 16 columns can be part of a Primary Key.
- Masked data will be created in the target without masking.
- OpenLDAP is not supported
- Replication from databases configured to use ADR (Accelerated Database Recovery) is not supported.
- Replicating a table or schema with an apostrophe (') in its name is not supported.
- Changes applied to the database using DACPAC will not be captured.
- MS-CDC Change Tables with fixed size columns (including NCHAR and CHAR data), the sum of which exceeds 8060 bytes, are not supported.
- System tables and views are not supported.
- Limitations when capturing DDL operations are as follows:
 - RENAME TABLE will not be captured. When a RENAME TABLE DDL is encountered, the task will continue capturing the changes without renaming the target table. If the task is stopped and resumed or if there is a recoverable error, the table will be suspended to allow the CT table to be manually aligned.
 - Table-level DDLs are not supported. When a table DDL is encountered, the DDL will be transferred to the target and the table will be suspended to allow the CT table to be manually aligned.
- Capturing DDL and DML statements within a single transaction is not supported.

Non-supported security features

Tables that use the following security features are not supported:

- Always Encrypted
- Row-Level Security

Unsupported data types

The following data types are not supported:

- CURSOR
- SQL_VARIANT
- TABLE



User-defined data types are supported according to their base-type. For example a user-defined data type based on DATETIME is handled as a DATETIME data type.

Required permissions

The user specified in the endpoint settings must have the following privileges on the source database:

Running with the db_owner role

If the user is a member of the db_owner database role, the following permissions must be granted:

- On the master database:
 - grant VIEW ANY DEFINITION
 - grant VIEW SERVER STATE
- On the MSDB database:
 - grant SELECT ON SYSJOBACTIVITY
 - grant SELECT ON SYSJOBS

Running without the db_owner role

If the user is not a member of the db_owner database role, the following permissions must be granted on the source database:

- grant VIEW DATABASE STATE
- grant SELECT



There are several limitations that apply when running without the db_owner role. For more information, see [Limitations and considerations](#) (page 327).

Supported compression methods

Supported compression methods

Row/Page Compression (at Partition Level)	Vardecimal Storage Format	Vardecimal Storage FormatSparse Columns	Vardecimal Storage FormatSparse ColumnsColumnar Structure Compression
Yes	No	No	No

Supported data types

The Microsoft Azure SQL (MS-CDC) source endpoint for Qlik Replicate supports most Microsoft Azure SQL Managed Instance data types. The following table shows the Microsoft Azure SQL Managed Instance source data types that are supported when using Qlik Replicate and the default mapping to Qlik Replicate data types. Note that Microsoft Azure SQL Managed Instance data types are only mapped to Qlik Replicate data types when the target endpoint is not Microsoft SQL Server-based. For information on data type mapping and collation support when the target endpoint is Microsoft SQL Server-based, see *Homogeneous replication (page 333)*.

For information on how to view the data type that is mapped in the target, see the section for the target endpoint you are using.

For additional information about Qlik Replicate data types, see [Replicate data types](#).



Collatable data types are indicated by an asterisk ().*

Supported data types

Microsoft Azure SQL Managed Instance	Qlik Replicate Data Types
BIGINT	INT8
BIT	BOOLEAN
DECIMAL	NUMERIC
INT	INT4
MONEY	NUMERIC (19,4)
NUMERIC (p,s)	NUMERIC
SMALLINT	INT2
SMALLMONEY	NUMERIC (10,4)
TINYINT	UINT1
REAL	REAL4
FLOAT	REAL8
DOUBLE	REAL8
DATETIME	DATETIME
DATETIME2	DATETIME
SMALLDATETIME	DATETIME
DATE	DATE
TIME	STRING (16)

Microsoft Azure SQL Managed Instance	Qlik Replicate Data Types
DATETIMEOFFSET	STRING
*CHAR	STRING
*VARCHAR	STRING
*VARCHAR (max) *TEXT To use this data type with Qlik Replicate, you must enable the use of CLOBs for a specific task. LOB columns for Microsoft Azure SQL (MS-CDC) tables are updated in the target even for UPDATE statements that did not change the value of the LOB column in Microsoft Azure SQL (MS-CDC). During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.	CLOB
*NCHAR	WSTRING
*NVARCHAR (length)	WSTRING
*NVARCHAR (max) *NTEXT To use this data type with Qlik Replicate, you must enable the use of NCLOBs for a specific task. LOB columns for Microsoft Azure SQL (MS-CDC) tables are updated in the target even for UPDATE statements that did not change the value of the LOB column in Microsoft Azure SQL (MS-CDC). During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.	NCLOB
BINARY	BYTES
VARBINARY	BYTES

Microsoft Azure SQL Managed Instance	Qlik Replicate Data Types
<p>VARBINARY (max)</p> <p>IMAGE</p> <p>LOB columns for Microsoft Azure SQL (MS-CDC) tables are updated in the target even for UPDATE statements that did not change the value of the LOB column in Microsoft Azure SQL (MS-CDC).</p> <p>To use this data type with Qlik Replicate, you must enable the use of BLOBs for a specific task.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p>	BLOB
TIMESTAMP	BYTES
UNIQUEIDENTIFIER	STRING
HIERARCHYID	VARCHAR (x)
<p>XML</p> <p>LOB columns for Microsoft Azure SQL (MS-CDC) tables are updated in the target even for UPDATE statements that did not change the value of the LOB column in Microsoft Azure SQL (MS-CDC).</p> <p>To use this data type with Qlik Replicate, you must enable the use of NCLOBs for a specific task.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p>	CLOB
GEOMETRY	CLOB
GEOGRAPHY	CLOB

Non-supported data types

Tables that include columns with the following data types are not supported by Qlik Replicate.

- CURSOR
- SQL_VARIANT
- TABLE



User-defined data types are supported according to their base-type. For example, a user-defined data type based on DATETIME is handled as a DATETIME data type.

Homogeneous replication

The following section describes how Replicate handles replication from Microsoft SQL Server (MS-CDC) to the following targets: Microsoft SQL Server, Microsoft Azure SQL Database, Microsoft Azure SQL Managed Instance, and Google Cloud SQL for SQL Server.

In homogeneous replication, source column and table collations will be replicated to the target as described in *Column and table collation* (page 333).



- In homogeneous replication, the source data first passes through the Qlik Replicate data type and is therefore subject to any limitations of that type.
- For information on Replicate data types and their limitations (where relevant), see [Replicate data types](#).
- To prevent data truncation when replicating columns with an **XML** data type, it is strongly recommended to enable the [Allow unlimited LOB size](#) option in the task settings.

Data type exceptions

When replicating from Microsoft SQL Server (MS-CDC) to a supported SQL Server-based target, source and target data types are identical, with the following exceptions:

Data type exceptions in homogeneous replication

Microsoft SQL Server (MS-CDC) Source	Microsoft SQL Server-based Target
VARCHAR	VARCHAR (x) (when x=0 or x>8000) else VARCHAR (max)
NVARCHAR (length)	NVARCHAR (x) (when x=0 or x>8000) else NVARCHAR (max)
VARBINARY	VARBINARY (x) (when x=0 or x>8000) else VARBINARY (max)
HIERARCHYID	VARCHAR (x)
TIMESTAMP	VARBINARY

Column and table collation

When replicating from Microsoft SQL Server (MS-CDC) to an SQL Server-based target, column and table collations will be replicated to the target.



To support collation replication, the DBA must ensure that the collations defined for Microsoft SQL Server (MS-CDC) are the same as those defined for the SQL Server-based target.

Non-nullable columns and Primary/Unique Index names

Primary/Unique Index names are preserved during homogeneous replication. Non-nullable columns are also preserved during homogeneous replication, with the exception of the following data types:

- text
- ntext1
- varchar(max)
- nvarchar(max)
- varbinary(max)
- image
- xml

Setting general connection properties

This section describes how to configure connection properties.

To do this:

1. In the Qlik Replicate Console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.
2. In the **Name** field, specify a name for your endpoint. This can be any name that will help to identify the endpoint being used.
3. In the **Description** field, optionally enter a description that helps to identify the endpoint.
4. Select **Source** as the endpoint **role**.
5. Select **Microsoft Azure SQL (MS-CDC)** as the endpoint **Type**.
6. Specify the **Server**. This is the host name or IP address of the computer with your Microsoft Azure SQL Managed Instance database.



To override the default port, add the port to the server name, separated by a comma. For example, if the server name is `myserver.company.local` and the port is `3333`, then the server name should be entered like this:
`myserver.company.local,3333`

7. Select one of the following authentication methods:
 - SQL Authentication
 - Azure Active Directory Authentication
 - Azure Active Directory Service Principal
8. In the **Username** field, enter your user name for accessing the database. When **Azure Active Directory Service Principal** is selected, this is the Client ID.

9. In the **Password** field, enter your password for accessing the database. When **Azure Active Directory Service Principal** is selected, this is the Secret Key.



To ensure that the password provided to Replicate is never used by any other principal, Replicate can be configured to automatically replace the user-entered password with a strong random password. For more information, see [Configuring Replicate to Automatically Replace the User-Entered Password](#).



- The username and password are case sensitive.
- To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**. If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box. To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

10. Type the **Database name** or click **Browse** and select one from the list of available databases. This is the name of the Microsoft Azure SQL Managed Instance database from where you are replicating the data.

Setting advanced connection properties

In the **Advanced** tab, you can enable CDC, provide internal parameters, set "More Options", and view a summary of your settings.

Enabling Change Data Capture

- **Enable MS-CDC on all captured tables** - Select this option to automatically enable MS-CDC for all selected source tables (on all columns). When this option is selected, the MS-CDC tables will be created in the database's default filegroup.
- **Capture DDL events:** When this option is selected, the following operations occur:
 - To facilitate capture of DDL events, Replicate creates triggers and other operational artifacts in the source database when the task starts. You can later remove these artifacts as described in *Removing Replicate artifacts from the database (page 347)*.
 - Streamed DDL events are captured.
- **Create DDL artifacts in schema** - The schema in which the operational DDL database artifacts will be created. The default value is "dbo".

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

Removing Replicate artifacts from the database

In order to capture DDLs, Qlik Replicate creates various artifacts in the Microsoft SQL Server database when the task starts. When the task completes, you may wish to remove these artifacts.

To remove the artifacts, execute the following statements (in the order they appear below), where `dbo` is the default schema in which the artifacts were created:



The event trigger does not belong to a specific schema.

```
DROP TRIGGER [attrep_mscdc_ddl_catcher] ON DATABASE
DROP TABLE [dbo].[attrep_mscdc_ddl_history]
```

9.17 Using Microsoft SQL Server (MS-CDC) as a source



This endpoint is supported from Replicate May 2022 Service Release 02 only.

This section describes how to set up Microsoft SQL Server (MS-CDC) as a source endpoint in a replication task. When replicating data from Microsoft SQL Server using MS-CDC, changes are captured from Microsoft Change Tables as opposed to the SQL Server transaction log. Eliminating the need to rely on the transaction log minimizes the risk of data loss resulting from online transaction log retention time and backup transaction log inaccessibility.

In this section:

- *Prerequisites (page 338)*
- *Limitations and considerations (page 339)*
- *Required permissions (page 329)*
- *Supported compression methods (page 329)*
- *Supported data types (page 342)*
- *Homogeneous replication (page 333)*
- *Setting general connection properties (page 345)*
- *Setting advanced connection properties (page 346)*
- *Removing Replicate artifacts from the database (page 347)*

Prerequisites

The following section lists the prerequisites for working with a Microsoft SQL Server (MS-CDC) source.

Client Prerequisites

Qlik Replicate for Windows

Install Microsoft ODBC Driver 18.1 for Windows on the Qlik Replicate Server machine.

Qlik Replicate for Linux

The procedure below assumes that you have installed a single default instance of Replicate on Linux (areplicate). If you have [installed multiple instances](#), replace areplicate with the name of the instance running the task with a Microsoft SQL Server source. If several instances are running such as task, the procedure needs to be repeated for each instance.

Install Microsoft ODBC Driver 18.1 for Linux on the Replicate machine.

Then, open a Unix shell and do the following:

1. Change the working directory to `<Replicate-Install-Dir>/bin`.
2. Stop the Replicate service:
`./areplicate stop`
3. Optionally, confirm that the service has stopped:
`./areplicate status`
4. Copy the driver location to the `site_arep_login.sh` file as follows:
`echo "export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/microsoft/msodbcsql18/lib64/" >> site_arep_login.sh`
This will add the driver to "LD_LIBRARY_PATH" and update the driver location in the **site_arep_login.sh** file.
5. Optionally, confirm that the driver location was copied:
`cat site_arep_login.sh`
6. Start the Replicate service:
`./areplicate start`
7. Optionally confirm that the service has started:
`./areplicate status`
8. Replicate requires the following ODBC library: `libmsodbcsql-18.1.so.1.1`
To check which library version is currently installed Issue the following command:
`ls /opt/microsoft/msodbcsql/lib64/`
If the existing library has a different version number (e.g. `libmsodbcsql-18.0.so.1.1`), you need to create a symbolic link between the existing library and the required library.
To do this, issue the following commands:
`cd /opt/microsoft/msodbcsql18/lib64/`
`ln -s existing_library_name libmsodbcsql-18.1.so.1.1`
where `existing_library_name` is the name of the currently installed library (e.g. `libmsodbcsql-18.0.so.1.1`).

Preparing the database for CDC

The following section explains how to set up the database for CDC.

To set up the database for CDC:

1. Set up the database for MS-CDC by executing the following:

```
exec source_db_name.dbo.sp_cdc_enable_db
```

where `source_db_name` is the name of the source database.

2. The replication task automatically enables MS-CDC for all the source tables. However, if you prefer to enable MS-CDC manually, execute the following script (for each of the intended source tables):

```
exec sys.sp_cdc_enable_table
```

```
@source_schema = N'schema_name',
```

```
@source_name = N'table_name',
```

```
@role_name = NULL
```

where `schema_name` and `table_name` are the names of the source schema and the source table respectively.



- The CT table names must use the default naming format: `<SchemaName>_<TableName>_CT`
- The CT table must include the following columns: `[__$start_lsn]`, `[__$end_lsn]`, `[__$seqval]`, `[__$operation]`, `[__$update_mask]`, `<all the data columns>`, `[__$command_id]`

3. Make sure the MS-CDC capture and cleanup jobs are running.



- You can control the jobs using `sys.sp_cdc_change_job`. For additional information, see the Microsoft Help.
- To reduce latency, you might want to set the `@pollinginterval` parameter to a small value.

TLS 1.2 prerequisites

If your environment matches the following:

- Microsoft SQL Server 2014
- Microsoft ODBC Driver 18 installed on Red Hat 8.1 or later

Then, in order to work with TLS 1.2, you must install a specific Service Pack on the Windows machine where Microsoft SQL Server is installed.

For more information, see <https://support.microsoft.com/en-us/help/3135244/tls-1-2-support-for-microsoft-sql-server>.

Limitations and considerations

You should be aware of the following limitations and considerations when using a Microsoft SQL Server (MS-CDC) source:

- Sparse tables are not supported.
- Replicating data from indexed views is not supported.

9 Adding and managing source endpoints

- Limitations due to the requirement to have MS-CDC enabled on all source tables include:
 - Rename column is not supported
 - Truncate table is not supported
 - WRITETEXT and UPDATETEXT utilities are not supported

For more information on MS-CDC limitations, see the Microsoft Help.

- Limitations when capturing DDL operations are as follows:
 - RENAME TABLE will not be captured. When a RENAME TABLE DDL is encountered, the task will continue capturing the changes without renaming the target table. If the task is stopped and resumed or if there is a recoverable error, the table will be suspended to allow the CT table to be manually aligned.
 - Table-level DDLs are not supported. When a table DDL is encountered, the DDL will be transferred to the target and the table will be suspended to allow the CT table to be manually aligned.
- When running without the db_owner role, the following limitations apply:
 - Capture of DDLs is not supported. To prevent replication errors, you must clear the **Capture DDL events** check box in [the endpoint settings' Advanced tab](#) before running the task.
 - MS-CDC cannot be enabled automatically and should therefore be enabled manually. To prevent replication errors, you must also clear the **Enable MS-CDC on all captured tables** check box in [the endpoint settings' Advanced tab](#) before running the task.



Special permissions are required for running without the db_owner role. For more information, see [Required permissions \(page 329\)](#).

- Replication of computed columns is supported in Full Load replication tasks only. During CDC, INSERTs to computed columns will appear as NULL on the target and UPDATES will be ignored.
- Microsoft Azure SQL Managed Instance partition switching is not supported.
- The following DML pattern is not supported:
`select <*> into <new_table> from <existing_table>`
- Column-level encryption is not supported.
- When inserting a value into Microsoft Azure SQL Managed Instance spatial data types (GEOGRAPHY and GEOMETRY), one can either ignore the SRID (Spatial Reference System Identifier) property - in which case the default SRID will be used (0 for GEOMETRY and 4326 for GEOGRAPHY) - or specify a different number. When replicating tables with spatial data types, Qlik Replicate replaces the SRID that was inserted by user with the default SRID.
- Replicating GEOGRAPHY and GEOMETRY data types is only supported in limited LOB mode.
- Columnstore indexes are not supported.
- Memory-optimized tables (using In-Memory OLTP) are not supported.
- Temporal databases are not supported
- Delayed durability is not supported
- Due to an ODBC limitation, no more than 16 columns can be part of a Primary Key.
- Masked data will be created in the target without masking.
- OpenLDAP is not supported

- Replication from databases configured to use ADR (Accelerated Database Recovery) is not supported.
- Capturing DDL and DML statements within a single transaction is not supported.
- Replicating a table or schema with an apostrophe (') in its name is not supported.
- Changes applied to the database using DACPAC will not be captured.
- MS-CDC Change Tables with fixed size columns (including NCHAR and CHAR data), the sum of which exceeds 8060 bytes, are not supported.
- The [AR_H_USER header column](#) is not supported.
- System tables and views are not supported.

Non-supported Microsoft SQL Server (MS-CDC) security features

Tables that use the following security features are not supported:

- Always Encrypted
- Row-Level Security

Required permissions

The user specified in the endpoint settings must have the following privileges on the source database:

Running with the db_owner role

If the user is a member of the db_owner database role, the following permissions must be granted:

- On the master database:
 - grant VIEW ANY DEFINITION
 - grant VIEW SERVER STATE
- On the MSDB database:
 - grant SELECT ON SYSJOBACTIVITY
 - grant SELECT ON SYSJOBS

Running without the db_owner role

If the user is not a member of the db_owner database role, the following permissions must be granted:

On the master database:

- grant VIEW DATABASE STATE

On the source database:

- grant VIEW DATABASE STATE
- grant SELECT

On the MSDB database:

- grant SELECT ON DBO.CDC_JOBS
- grant SELECT ON DBO.SYSJOBACTIVITY
- grant SELECT ON DBO.SYSJOBS



There are several limitations that apply when running without the `db_owner` role. For more information, see *Limitations and considerations* (page 339).

Supported data types

The following table shows the source data types that are supported when using Qlik Replicate and the default mapping to Qlik Replicate data types. Note that source data types are only mapped to Qlik Replicate data types when the target endpoint is not Microsoft SQL Server-based. For information on data type mapping and collation support when the target endpoint is Microsoft SQL Server-based, see *Homogeneous replication* (page 333).

For information on how to view the data type that is mapped in the target, see the section for the target endpoint you are using.



For additional information about Qlik Replicate data types, see [Replicate data types](#).





Collatable data types are indicated by an asterisk (*).

Supported data types

Microsoft SQL Server (MS-CDC)	Qlik Replicate Data Types
BIGINT	INT8
BIT	BOOLEAN
DECIMAL	NUMERIC
INT	INT4
MONEY	NUMERIC (19,4)
NUMERIC (p,s)	NUMERIC
SMALLINT	INT2
SMALLMONEY	NUMERIC (10,4)
TINYINT	UINT1
REAL	REAL4
FLOAT	REAL8
DOUBLE	REAL8
DATETIME	DATETIME
DATETIME2	DATETIME
SMALLDATETIME	DATETIME
DATE	DATE

Microsoft SQL Server (MS-CDC)	Qlik Replicate Data Types
TIME	STRING (16)
DATETIMEOFFSET	STRING
*CHAR	STRING
*VARCHAR	STRING
*VARCHAR (max) *TEXT	CLOB
<div style="display: flex; align-items: flex-start;"> <div style="margin-right: 10px; text-align: center;">  </div> <ul style="list-style-type: none"> • <i>LOB columns for Microsoft SQL Server (MS-CDC) tables are updated in the target even for UPDATE statements that did not change the value of the LOB column in Microsoft SQL Server (MS-CDC).</i> • <i>To use this data type with Qlik Replicate, you must enable the use of CLOBs for a specific task.</i> • <i>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</i> </div>	
*NCHAR	WSTRING
*NVARCHAR (length)	WSTRING
*NVARCHAR (max) *NTEXT	NCLOB
<div style="display: flex; align-items: flex-start;"> <div style="margin-right: 10px; text-align: center;">  </div> <ul style="list-style-type: none"> • <i>LOB columns for Microsoft SQL Server (MS-CDC) tables are updated in the target even for UPDATE statements that did not change the value of the LOB column in Microsoft SQL Server (MS-CDC).</i> • <i>To use this data type with Qlik Replicate, you must enable the use of NCLOBs for a specific task.</i> • <i>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</i> </div>	
BINARY	BYTES
VARBINARY	BYTES

Microsoft SQL Server (MS-CDC)	Qlik Replicate Data Types
VARBINARY (max) IMAGE <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;">  <ul style="list-style-type: none"> <i>LOB columns for Microsoft SQL Server (MS-CDC) tables are updated in the target even for UPDATE statements that did not change the value of the LOB column in Microsoft SQL Server (MS-CDC).</i> <i>To use this data type with Qlik Replicate, you must enable the use of BLOBs for a specific task.</i> <i>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</i> </div>	BLOB
TIMESTAMP	BYTES
UNIQUEIDENTIFIER	STRING
HIERARCHYID	VARCHAR (x)
XML <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;">  <ul style="list-style-type: none"> <i>LOB columns for Microsoft SQL Server (MS-CDC) tables are updated in the target even for UPDATE statements that did not change the value of the LOB column in Microsoft SQL Server (MS-CDC).</i> <i>To use this data type with Qlik Replicate, you must enable the use of NCLOBs for a specific task.</i> <i>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</i> </div>	CLOB
GEOMETRY	CLOB
GEOGRAPHY	CLOB

Non-supported data types

Tables that include columns with the following data types are not supported by Qlik Replicate.

- CURSOR
- SQL_VARIANT
- TABLE



User-defined data types are supported according to their base-type. For example, a user-defined data type based on DATETIME is handled as a DATETIME data type.

Setting general connection properties

This section describes how to configure connection properties.

To do this:

1. In the Qlik Replicate Console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints* (page 114).
2. In the **Name** field, specify a name for your endpoint. This can be any name that will help to identify the endpoint being used.
3. In the **Description** field, optionally enter a description that helps to identify the endpoint.
4. Select **Source** as the endpoint **role**.
5. Select **Microsoft SQL Server (MS-CDC)** as the endpoint **Type**.
6. Specify the **Server name**. This is the host name or IP address of the computer with your Microsoft SQL Server database.



To override the default port, add the port to the server name, separated by a comma. For example, if the server name is `myserver.company.local` and the port is `3333`, then the server name should be entered like this:
`myserver.company.local,3333`

7. To connect to the database using Windows authentication, Select **Use Windows authentication** (only relevant when Replicate is installed on Windows). When not selected, SQL Server authentication will be used instead.


If you select **Use Windows authentication**, Replicate will use a trusted connection to connect with SQL Server using the LogOn account set up for the Qlik Replicate Server service – no explicit username nor password are needed. This mode must be configured in the Microsoft SQL Server database by the database administrator. Note that this option is not relevant when Replicate Server is running on Linux.




When using **Windows authentication**, make sure that the user account that is associated with the **Qlik Replicate Server** service has Network read and write permissions. This must be configured by a Windows system administrator.


See also *Working with Windows Authentication* (page 347).

If you did not select **Use Windows authentication**, specify a **Username** and **Password** for connecting to Microsoft SQL Server. The user name and password can be of a domain user or of a database user, depending on the configuration of Microsoft SQL Server. If you do not know this information, see your Microsoft SQL Server Database administrator.

 To ensure that the password provided to Replicate is never used by any other principal, Replicate can be configured to automatically replace the user-entered password with a strong random password. For more information, see [Configuring Replicate to Automatically Replace the User-Entered Password](#).



- This information is case sensitive.
- To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**. If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box. To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

 Make sure that the Microsoft SQL Server user has the correct access privileges. For information on how to provide the required privileges, see [Required permissions \(page 329\)](#).

8. Type the **Database name** or click **Browse** and select one from the list of available databases. This is the name of the Microsoft Azure SQL Managed Instance database from where you are replicating the data.

Setting advanced connection properties

In the **Advanced** tab, you can enable CDC, provide internal parameters, set "More Options", and view a summary of your settings.

Enabling Change Data Capture

- **Enable MS-CDC on all captured tables** - Select this option to automatically enable MS-CDC for all selected source tables (on all columns). When this option is selected, the MS-CDC tables will be created in the database's default filegroup.
- **Capture DDL events:** When this option is selected, the following operations occur:
 - To facilitate capture of DDL events, Replicate creates triggers and other operational artifacts in the source database when the task starts. You can later remove these artifacts as described in [Removing Replicate artifacts from the database \(page 347\)](#).
 - Streamed DDL events are captured.
- **Create DDL artifacts in schema** - The schema in which the operational DDL database artifacts will be created. The default value is "dbo".

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

Working with Windows Authentication

You can configure the Qlik Replicate Microsoft Azure SQL (MS-CDC) endpoint to log in to Microsoft Azure SQL (MS-CDC) (on Windows) using Windows authentication.

If you choose this option, you also need to make sure that:

- The Microsoft Azure SQL (MS-CDC) instance is set up to allow Windows log on.
- The Qlik Replicate user is specified as the "Log on as" user for the "Qlik Replicate Server" service account.
-OR-
Microsoft Azure SQL (MS-CDC) is configured to allow login for the Qlik Replicate Server service account.

Removing Replicate artifacts from the database

In order to capture DDLs, Qlik Replicate creates various artifacts in the Microsoft SQL Server database when the task starts. When the task completes, you may wish to remove these artifacts.

To remove the artifacts, execute the following statements (in the order they appear below), where dbo is the default schema in which the artifacts were created:



The event trigger does not belong to a specific schema.

```
DROP TRIGGER [attrep_mscdc_ddl_catcher] ON DATABASE  
DROP TABLE [dbo].[attrep_mscdc_ddl_history]
```

9.18 Using Microsoft SQL Server as a source

This section describes how to set up and use a Microsoft SQL Server database as the source database in a replication task.

In this section:

- *Supported editions (page 349)*
- *Prerequisites (page 349)*
- *Limitations and considerations (page 350)*
- *Working with Microsoft SQL Server AlwaysOn Availability Groups (page 352)*
- *Required permissions (page 354)*
- *Supported compression methods (page 354)*
- *Microsoft SQL Server source data types (page 355)*
- *Homogeneous replication (page 358)*
- *Preparing the Microsoft SQL Server database for CDC (page 360)*
- *Setting general connection properties (page 362)*
- *Setting advanced connection properties (page 364)*
- *Working with Windows authentication (page 367)*
- *Artifacts created by Replicate in the database (page 367)*

Supported editions

The following Microsoft SQL Server editions are supported:

- Enterprise Edition
- Standard Edition
- Workgroup Edition
- Developer Edition
- SQL Server Web Edition (supported with Full Load tasks only)

Prerequisites

Make sure that the following prerequisites have been met:

Client prerequisites

Qlik Replicate for Windows

Install Microsoft ODBC Driver 18.1 for Windows on the Qlik Replicate Server machine.

Qlik Replicate for Linux

The procedure below assumes that you have installed a single default instance of Replicate on Linux (areplicate). If you have [installed multiple instances](#), replace areplicate with the name of the instance running the task with a Microsoft SQL Server source. If several instances are running such as task, the procedure needs to be repeated for each instance.

Install Microsoft ODBC Driver 18.1 for Linux on the Replicate machine.

Then, open a Unix shell and do the following:

1. Change the working directory to `<Replicate-Install-Dir>/bin`.
2. Stop the Replicate service:

```
./areplicate stop
```
3. Optionally, confirm that the service has stopped:

```
./areplicate status
```
4. Copy the driver location to the `site_arep_login.sh` file as follows:

```
echo "export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/microsoft/msodbcsql18/lib64/" >>  
site_arep_login.sh
```

This will add the driver to "LD_LIBRARY_PATH" and update the driver location in the **site_arep_login.sh** file.
5. Optionally, confirm that the driver location was copied:

```
cat site_arep_login.sh
```
6. Start the Replicate service:

```
./areplicate start
```
7. Optionally confirm that the service has started:

```
./areplicate status
```
8. Replicate requires the following ODBC library: `libmsodbcsql-18.1.so.1.1`
To check which library version is currently installed Issue the following command:

```
ls /opt/microsoft/msodbcsql/lib64/
```

If the existing library has a different version number (e.g. `libmsodbcsql-18.0.so.1.1`), you need to create a symbolic link between the existing library and the required library.

To do this, issue the following commands:

```
cd /opt/microsoft/msodbcsql18/lib64/  
ln -s existing_library_name libmsodbcsql-18.1.so.1.1
```

where *existing_library_name* is the name of the currently installed library (e.g. `libmsodbcsql-18.0.so.1.1`).

General prerequisites

- The Microsoft SQL Server database must be enabled for MS-REPLICATION. For more information, see *Setting up Microsoft SQL Server for replication* (page 361).
- A Microsoft SQL Server account with the specific access privileges is required. For more information, see *Required permissions* (page 354).
- Microsoft SQL Server as a source must be configured for a full backup to work with Replicate. For more information, see *Preparing Microsoft SQL Server backup and recovery* (page 360).
- When decompressing a compressed backup transaction log, Replicate writes the decompressed log to the same location as the compressed log. Consequently, the location of the compressed backup transaction log must have enough space to contain the decompressed log as well.

TLS 1.2 prerequisites

If your environment matches the following:

- Microsoft SQL Server 2014
- Microsoft ODBC Driver 18 installed on Red Hat 8.1 or later

Then, in order to work with TLS 1.2, you must install a specific Service Pack on the Windows machine where Microsoft SQL Server is installed.

For more information, see <https://support.microsoft.com/en-us/help/3135244/tls-1-2-support-for-microsoft-sql-server>.

Limitations and considerations

You should be aware of the following limitations and considerations when using a Microsoft SQL Server source:

- The Microsoft SQL Server Replication Publisher definitions for the database that was used in the task are not removed if you delete the task. A Microsoft SQL Server system administrator must delete these definitions from Microsoft SQL Server.
- Replication of Sparse tables is supported in "Full Load" only replication mode.
- Replication of data from indexed views is not supported.
- Renaming tables using `sp_rename` is not supported (e.g. `sp_rename 'Sales.SalesRegion', 'SalesReg';`)
- Renaming columns using `sp_rename` is not supported (e.g. `sp_rename 'Sales.Sales.Region', 'RegID', 'COLUMN';`)
- TRUNCATE events will not be captured.

9 Adding and managing source endpoints

- Replication of computed columns is supported in Full Load replication tasks only. During CDC, INSERTs to computed columns will appear as NULL on the target and UPDATES will be ignored.
- Microsoft SQL Server partition switching is not supported.
- When using the WRITETEXT and UPDATETEXT utilities, events applied to the source database will not be captured.
- The following DML pattern is not supported:

```
select <*> into <new_table> from <existing_table>
```
- Column-level encryption is not supported.
- The following limitations apply when accessing the backup transaction logs:
 - Encrypted backups are not supported.
 - Backups stored at a URL or on Windows Azure are not supported.
- The following limitations apply when accessing the backup transaction logs *at file level*:
 - The backup transaction logs must reside in a shared folder with the appropriate permissions and access rights.
 - Active transaction logs are accessed through the Microsoft SQL Server API (and not at file-level).
 - The Qlik Replicate and Microsoft SQL Server machines must reside in the same domain.
 - Transparent Data Encryption (TDE) is only supported when accessing the backup transaction logs using SQL Server's native functionality. It is not supported when using the endpoint's file-level access option.
 - Unix platforms are not supported.
 - Reading the backup logs from multiple stripes is not supported.
 - The File Stream and Full Text features are not supported when the endpoint's [Replicate has file-level access to the backup log files](#) option is enabled.

For more information on configuring Qlik Replicate to access the backup transaction logs at file-level access, see *Setting advanced connection properties* (page 364).

- Microsoft SQL Server backup to multiple disks (i.e. MIRROR TO DISK) is not supported.
- When inserting a value into SQL Server spatial data types (GEOGRAPHY and GEOMETRY), one can either ignore the SRID (Spatial Reference System Identifier) property - in which case the default SRID will be used (0 for GEOMETRY and 4326 for GEOGRAPHY), or specify a different number. When replicating tables with spatial data types, the SRID that was inserted by user will be replaced with the default SRID.
- Replicating GEOGRAPHY and GEOMETRY data types is only supported in limited LOB mode.
- If your database is not set up for MS-REPLICATION or MS-CDC, you can still capture tables that do not have a Primary Key, but bear in mind that in such a setup only INSERT/DELETE DML events will be captured. UPDATE events will be ignored.
- Columnstore indexes are not supported.
- Memory-optimized tables (using In-Memory OLTP) are not supported.
- Temporal tables are not supported
- Delayed durability is not supported
- Table change tracking is not supported
- Masked data will be created in the target without masking.
- OpenLDAP is not supported

- Replication directly from a secondary Microsoft SQL Server database is not supported. Consequently, the Microsoft SQL Server endpoint must be configured to connect to the primary database or the AlwaysOn Listener.
- Replication from databases configured to use ADR (Accelerated Database Recovery) is not supported.
- Capturing DDL and DML statements within a single transaction is not supported.
- Replicating a table or schema with an apostrophe (') in its name is not supported.
- UPDATES to a Primary Key/Unique Index that affect multiple rows may cause conflicts when applying the changes to the target. This may occur, for example, if the UPDATES are applied to the target as INSERT/DELETE operations rather than a single UPDATE operation. In such a scenario (where UPDATES to a Primary Key/Unique Index affect multiple rows), working in Batch optimized apply Change Processing mode is not supported as it may result in the table being ignored. Working in Transactional Apply Change Processing mode however, may result in constraint violations. If this happens, you either need to reload the relevant table or locate the problematic records in the *Apply exceptions* (page 1298) Control Table and edit them manually in the target database. See also: *Change Processing Tuning* (page 1181).
- Changes applied to the database using DACPAC will not be captured.
- CDC is not supported from tables which have both compressed and uncompressed partitions. If only some of the partitions are compressed, the following message will be shown: **Table 'xxx' is not uniformly mapped across partitions. Therefore it is excluded from CDC.**
- System tables and views are not supported.

Unsupported Microsoft SQL Server security features

Tables that use the following Microsoft SQL Server security features are not supported:

- Always Encrypted
- Row-Level Security

Unsupported data types

The following data types are not supported:

- CURSOR
- SQL_VARIANT
- TABLE



User-defined data types are supported according to their base-type. For example a user-defined data type based on DATETIME is handled as a DATETIME data type.

Working with Microsoft SQL Server AlwaysOn Availability Groups

The Microsoft SQL Server AlwaysOn Availability Groups feature is a high-availability, disaster-recovery solution that provides an enterprise-level alternative to database mirroring.

Prerequisites

- Credentials used for connecting to individual replicas should be identical to those used for connecting to the AlwaysOn Listener.
- Replicas should be configured to allow connections.

To configure this, the DBA should run the following script:

```
ALTER AVAILABILITY GROUP [{ag_name}]  
MODIFY REPLICA ON '{replica_name}'  
WITH (SECONDARY_ROLE (ALLOW_CONNECTIONS = ALL, READ_ONLY_ROUTING_URL =  
'TCP://{replica_name}:{Port}'))
```

Configuring Replicate to access the AlwaysOn Listener

When working with AlwaysOn Availability Groups, you need to specify the IP address/host name and access credentials of the AlwaysOn Listener in the connection properties [Microsoft SQL Server source endpoint settings](#).

Limitations

DAG (Distributed Availability Groups) is not supported.

Accessing backup logs in AlwaysOn Availability Groups

As opposed to active transaction logs which are synchronized across the AlwaysOn Availability Group, backup transaction logs are different for each individual replica. Consequently, when Replicate needs to access the backup transaction logs, it "asks" each Replica in turn if it has the required logs, until the backup transaction logs are located. Note that if one of the Replicas that Replicate is trying to reach is offline, Replicate will wait until that Replica comes back online and then query it for the backup transaction logs.

Replicate needs to access the backup transaction logs on each Replica (or until it finds the required logs) in the following scenarios:

- Working in backup only mode.
For more information on this mode, see [Microsoft SQL Server source endpoint settings](#).
- Starting a task from a specific timestamp.
For more information on this option, see the **Tables are already loaded** option in *Advanced Run Options (page 1205)*.
- Due to latency i.e. if there is a high rate of events (changes) that Replicate is unable to process using the active log only.





*Reading backup logs from AlwaysOn Availability Groups requires opening connections to all of the replicas. If you do not want to do this, set the **Change processing mode** in the **Advanced** tab to **Online Logs only**.*

Setting up a non-sysadmin user when working with AlwaysOn availability groups

This section explains how to support a non-sysadmin user when working with AlwaysOn availability groups. For information on setting up non-sysadmin in a standard environment, see *Required permissions (page 354)*.

The setup procedure involves running scripts. As copying the scripts from the PDF introduces formatting issues that invalidate the scripts, the instructions are only accessible from the HTML-based online Help. To access the instructions, click the **Help** link in the top right of the Replicate Console and navigate to **Setting up a Non-Sysadmin User when Working with AlwaysOn Availability Groups**.

1.  *MS-REPLICATION distributor should either be configured as local or in a way that allows access to non-sysadmin users via the associated linked server.*
2. If the [Exclusive automatic truncation](#) endpoint option is enabled, stop the MS-REPLICATION Log Reader job.
3.  *The login should be created with the same SID on each replica.*


Required permissions

To use Microsoft SQL Server as a source in a replication task, the user specified in the Microsoft SQL Server endpoint [connection settings](#) must be one of the following:

- A member of both the `db_owner` database role and the `sysAdmin` fixed server role.
- A member of the `db_owner` database role but not a member of `sysAdmin` fixed server role. This also requires you to perform the procedures described in **Setting up a non-sysadmin user in a standalone environment** below.

Setting up a non-sysadmin user in a standalone environment

The procedure for setting up a non-sysadmin includes running scripts. As copying the scripts from the PDF introduces formatting issues that invalidate the scripts, the instructions are only accessible from the HTML-based online Help. To access the instructions, click the **Help** link in the top right of the Replicate Console and navigate to **Setting up a Non-Sysadmin User**.

 *If you wish to set up a non-sysadmin user in an AlwaysOn environment, see [Setting up a non-sysadmin user when working with AlwaysOn availability groups](#) (page 353).*

Supported compression methods

The table below lists which compression methods are supported for each Microsoft SQL Server version.

Supported compression methods

Microsoft SQL Server Version	Row/Page Compression (at Partition Level)	Vardecimal Storage Format	Vardecimal Storage Format Sparse Columns	Vardecimal Storage Format Sparse Columns Columnar Structure Compression
2012-2019	Yes	No	No	No

Microsoft SQL Server source data types

The Microsoft SQL Server source for Qlik Replicate supports most Microsoft SQL Server data types. The following table shows the Microsoft SQL Server source data types that are supported when using Qlik Replicate and the default mapping to Qlik Replicate data types. Note that Microsoft SQL Server data types are only mapped to Qlik Replicate data types when the target endpoint is not Microsoft SQL Server. For information on data type mapping and collation support when the target endpoint is Microsoft SQL Server, see *Homogeneous replication (page 358)* below.

For information on how to view the data type that is mapped in the target, see the section for the target endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.



When row size exceeds 8060, the VARCHAR, NVARCHAR, and VARBINARY data types will be stored out-of-row. To replicate such data, use the source lookup function described in *Data Enrichment (page 1157)*.



Collatable data types are indicated by an asterisk (*).

Supported source data types

Microsoft SQL Server Data Types	Qlik Replicate Data Types
BIGINT	INT8
BIT	BOOLEAN
DECIMAL	NUMERIC
INT	INT4
MONEY	NUMERIC (19,4)
NUMERIC (p,s)	NUMERIC
SMALLINT	INT2
SMALLMONEY	NUMERIC (10,4)

9 Adding and managing source endpoints

Microsoft SQL Server Data Types	Qlik Replicate Data Types
TINYINT	UINT1
REAL	REAL4
FLOAT	REAL8
DOUBLE	REAL8
DATETIME	DATETIME
DATETIME2	DATETIME
SMALLDATETIME	DATETIME
DATE	DATE
TIME	STRING (16)
DATETIMEOFFSET	STRING
*CHAR	STRING
*VARCHAR	STRING
*VARCHAR (max) *TEXT To use this data type with Qlik Replicate, you must enable the use of CLOBs for a specific task. LOB columns for Microsoft SQL Server tables are updated in the target even for UPDATE statements that did not change the value of the LOB column in Microsoft SQL Server. During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.	CLOB
*NCHAR	WSTRING
*NVARCHAR (length)	WSTRING

Microsoft SQL Server Data Types	Qlik Replicate Data Types
<p>*NVARCHAR (max)</p> <p>*NTEXT</p> <p>To use this data type with Qlik Replicate, you must enable the use of NCLOBs for a specific task.</p> <p>LOB columns for Microsoft SQL Server tables are updated in the target even for UPDATE statements that did not change the value of the LOB column in Microsoft SQL Server.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p>	NCLOB
BINARY	BYTES
VARBINARY	BYTES
<p>VARBINARY (max)</p> <p>IMAGE</p> <p>LOB columns for Microsoft SQL Server tables are updated in the target even for UPDATE statements that did not change the value of the LOB column in Microsoft SQL Server.</p> <p>To use this data type with Qlik Replicate, you must enable the use of BLOBs for a specific task.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p>	BLOB
TIMESTAMP	BYTES
UNIQUEIDENTIFIER	STRING
HIERARCHYID	VARCHAR (x)

Microsoft SQL Server Data Types	Qlik Replicate Data Types
<p>XML</p> <p>LOB columns for Microsoft SQL Server tables are updated in the target even for UPDATE statements that did not change the value of the LOB column in Microsoft SQL Server.</p> <p>To use this data type with Qlik Replicate, you must enable the use of NCLOBs for a specific task.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p>	CLOB
GEOMETRY	CLOB
GEOGRAPHY	CLOB

Unsupported data types

The following data types are not supported:

- CURSOR
- SQL_VARIANT
- TABLE



User-defined data types are supported according to their base-type. For example a user-defined data type based on DATETIME is handled as a DATETIME data type.

Homogeneous replication

The following section describes how Replicate handles replication from Microsoft SQL Server to the following targets: Microsoft SQL Server, Microsoft Azure SQL Database, Microsoft Azure SQL Managed Instance, and Google Cloud SQL for SQL Server.

In homogeneous replication, source column and table collations will be replicated to the target as described in *Column and table collation* (page 359).



In homogeneous replication, the source data first passes through the Qlik Replicate data type and is therefore subject to any limitations of that type.

For information on Replicate data types and their limitations (where relevant), see [Replicate data types](#) (page 81).

For information on which Replicate data types the source data passes through when replicating from Microsoft SQL Server, see the [Microsoft SQL Server to Qlik Replicate data types mapping table](#) described earlier.



*To prevent data truncation when replicating columns with an **XML** data type, it is strongly recommended to enable the [Allow unlimited LOB size](#) option in the task settings.*

Data type exceptions

When replicating from Microsoft SQL Server database to an SQL Server-based target, source and target data types are identical for all supported Microsoft SQL Server versions, with the following exceptions:

Data type exceptions in homogeneous replication

Microsoft SQL Server Source	SQL Server-based Target
VARCHAR	VARCHAR (x) (when x=0 or x>8000) else VARCHAR (max)
NVARCHAR (length)	NVARCHAR (x) (when x=0 or x>8000) else NVARCHAR (max)
VARBINARY	VARBINARY (x) (when x=0 or x>8000) else VARBINARY (max)
HIERARCHYID	VARCHAR (x)
TIMESTAMP	VARBINARY

Column and table collation

When replicating from Microsoft SQL Server to an SQL Server-based target, column and table collations will be replicated to the target.



To support collation replication, the DBA must ensure that the collations defined for the source Microsoft SQL Server database are the same as those defined for the SQL Server-based target database.

Non-nullable columns and primary/unique index names

Primary/Unique Index names are preserved during homogeneous replication. Non-nullable columns are also preserved during homogeneous replication, with the exception of the following data types:

- text
- ntext1
- varchar(max)
- nvarchar(max)
- varbinary(max)
- image
- xml

Preparing the Microsoft SQL Server database for CDC

This topics describes the configuration requirements for using a Microsoft SQL Server database. A Microsoft SQL Server system administrator should carry out these tasks.

- *Defining Microsoft SQL Server database settings (page 362)*
- *Preparing Microsoft SQL Server backup and recovery (page 360)*
- *Setting up Microsoft SQL Server for replication (page 361)*
- *Replicating tables that do not have a primary key (page 361)*

Preparing Microsoft SQL Server backup and recovery

Changes are captured from the database transaction log (TLOG). The TLOG is maintained by Microsoft SQL Server for recovery purposes. All changes made to a database are written to the TLOG. The following happens when recovery is required:

- A backup copy of the database is made.
- Logged events are taken and used in a rolling-forward process where the recorded changes are replayed against that copy.

To prepare for backup and recovery you must make sure that the Microsoft SQL Server Recovery Model is set up. You select the Recovery Model in the Microsoft SQL Server Management Studio. This should be carried out by a Microsoft SQL Server system administrator.

The TLOG data is truncated as soon as it is no longer needed. Therefore, the TLOG is not persistent. However, persistency of the changed data can be ensured by doing the following:

- Perform a full database backup before starting the task
- Set the Recovery Model to **Bulk logged** or **Full**.

To set the recovery model:

In the database properties **Options** tab, set the Recovery Model to **Bulk logged** or **Full**. In these modes, the transaction log is more durable.



After setting the Recovery Model, it is strongly recommended not to change it; doing so may result in loss of data.

Setting up Microsoft SQL Server for replication

You need to enable your Microsoft SQL Server database for MS-REPLICATION.

In the Microsoft SQL Server's Management Studio, follow the instructions provided by the Configure Distribution wizard to set up replication or see the Microsoft SQL Server documentation.

To open the wizard from Microsoft SQL Server:

1. In the Microsoft SQL Server Management Studio, right-click the Replication folder and select **Configure Distribution**.
The Configure Distribution wizard opens.
2. Make the following selections:
In the Distributor step, select **<Microsoft SQL Server Name> will act as its own distributor; Microsoft SQL Server will create a distribution database and log.**

Replicating tables that do not have a primary key



This functionality is supported only for Microsoft SQL Server Enterprise edition.

By default, Qlik Replicate automatically sets up MS-REPLICATION for each of the source tables in a replication task. However, MS-REPLICATION requires each of the source tables to have a primary key, which may not always be the case. Therefore, if you need to replicate tables that do not have a primary key, the following options are available:

- Use MS-CDC (page 361)
- Do not use MS-Replication or MS-CDC (page 362)

Use MS-CDC

To set up MS-CDC, you first need to enable MS-CDC for the database by running the following command:

```
use [DBname]
EXEC sys.sp_cdc_enable_db
```

Then you need to enable MS-CDC for each of the source tables by running the following command:

```
EXECUTE sys.sp_cdc_enable_table @source_schema = N'Myschema', @source_name = N'MyTable',
@role_name = NULL;
```



Replicating tables that do not have a Primary Key or a Unique Index may adversely affect performance (since additional database resources are required to capture the changes). However, you can prevent performance issues related to the absence of Primary Keys or a Unique Index by manually adding indexes to the target tables.

For more information on setting up MS-CDC for specific tables, please refer to the Microsoft website.

Do not use MS-Replication or MS-CDC

If your database is not set up for MS-REPLICATION or MS-CDC, you can still capture tables that do not have a Primary Key, but bear in mind that in such a setup only INSERT/DELETE DML events will be captured. UPDATE events will be ignored.

It is also important to note that a DELETE statement executed on an UPDATED source record, will not be applied on the target.

Defining Microsoft SQL Server database settings

Set the following for the Microsoft SQL Server database(s) you are using as a source:

- From the Object Explorer in the Microsoft SQL Server Management Studio, right click the database and select **Properties**. In the **Options** tab, set the **Recovery model** to **Bulk logged** or **Full**. In this mode, the transaction Log is more durable and truncation occurs less frequently.
- Ensure that there is a full database backup for each Microsoft SQL Server database that you are using as a source.
- When creating a connection string, it is possible to use any parameter supported by Microsoft SQL Server. The Microsoft SQL Server system administrator must ensure that the Microsoft SQL Server instance is configured correctly so that the proper authentication credentials are accepted.
- To be able to work with MS-REPLICATION, each of the source tables must have a primary key.

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a Microsoft SQL Server source endpoint to Qlik Replicate:

1. In the Qlik Replicate Console, click Manage Endpoint Connections to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Microsoft SQL Server database. This is optional.
4. Select **Source** as the database **role**.
5. Select **Microsoft SQL Server** as the database **Type**.

- Specify the **Server name**. This is the host name or IP address of the computer with the Microsoft SQL Server instance containing the source database.



- When working with AlwaysOn Availability Groups, you need to specify the IP address/host name and access credentials of the AlwaysOn Listener
- To override the default port, add the port to the server name, separated by a comma. For example, if the server name is `myserver.company.local` and the port is `3333`, then the server name should be entered like this:
`myserver.company.local,3333`

- Select **Windows authentication** (only relevant when Replicate is installed on Windows) or **SQL Server authentication**.

If you select **Windows authentication**, the Qlik Replicate Server will use a trusted connection to connect with SQL Server using the LogOn account set up for the Qlik Replicate Server service – no explicit username nor password are needed. This mode must be configured in the Microsoft SQL Server database by the database administrator. Note that this option is not relevant when Replicate Server is running on Linux.



When using **Windows authentication**, make sure that the user account that is associated with the **Qlik Replicate Server** service has Network read and write permissions. This must be configured by a Windows system administrator.

See also *Working with Windows authentication* (page 367).

If you select **SQL Server authentication**, you must specify a username and password for connecting to Microsoft SQL Server. The username and password may be of a domain user or of a database user, depending on the configuration of Microsoft SQL Server. If you do not know this information, see your Microsoft SQL Server Database Administrator.



To ensure that the password provided to Replicate is never used by any other principal, Replicate can be configured to automatically replace the user-entered password with a strong random password. For more information, see [Configuring Replicate to Automatically Replace the User-Entered Password](#).



- This information is case sensitive.
- To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**. If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box. To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.



Make sure that the Microsoft SQL Server user has the correct access privileges. For information on how to provide the required privileges, see [Required permissions \(page 354\)](#).

8. Type the **Database name** or click **Browse** and select one from the list of available databases. This is the name of the database from where you are replicating the data.

Setting advanced connection properties

In the **Advanced** tab, you can set the following properties:

- **Prevent truncation of unread changes from TLOG:** For optimal performance, Qlik Replicate will try to capture all unread changes from the active transaction log (TLOG). However, sometimes due to truncation, the active TLOG may not contain all of the unread changes. When this occurs, Qlik Replicate accesses the backup log to capture the missing changes. To minimize the need to access the backup log, Qlik Replicate prevents truncation using one of the following methods:
 - **Start transactions in the database:** This is the default method. When this method is used, Qlik Replicate prevents TLOG truncation by mimicking a transaction in the database. As long as such a transaction is open, changes that appear after the transaction started will not be truncated. If you need Microsoft Replication to be enabled in your database, then you must choose this method.



This method also requires the Log Reader Agent to be running to enable truncation of the Microsoft SQL Server active transaction log. Note that if the Log Reader Agent is not running, the active log may become full, causing the source database to be essentially "read-only" until the issue is resolved.



*When this option is selected, Replicate creates a table named `attrep_truncation_safeguard` in the source database. This is a very small but important table whose purpose is to prevent truncation of the transaction log by mimicking a transaction in the database. Make sure that the table is not included any maintenance plan as it may cause the maintenance job to fail. The table can be safely deleted if there are no tasks configured with the **Start transactions in the database** option.*

- **Exclusively use sp_repldone within a single task:** When this method is used, Qlik Replicate reads the changes and then uses `sp_repldone` to mark the TLOG transactions as ready for truncation. Although this method does not involve any transactional activities, it can only be used when Microsoft Replication is not running. Also, using this method, only one Qlik Replicate task can access the database at any given time. Therefore, if you need to run parallel Qlik Replicate tasks against the same database, use the default method.



- *This method requires the Log Reader Agent to be stopped in the database. If the Log Reader Agent is running when the task starts, Qlik Replicate will forcibly stop it. Alternatively, you can stop the Log Reader Agent manually, before starting the Qlik Replicate task. For instructions on how to do this, refer to the Microsoft SQL Server Management Studio help.*
- *When using this method with MS-CDC, the CDC Capture and CDC Cleanup jobs should be stopped and disabled.*
- *This method is not available when the Microsoft SQL Server Replication job resides on a remote Distributor machine as Replicate does not have access to the remote machine.*

- **Apply TLOG truncation prevention policy every (seconds):** Specify how often to prevent TLOG truncation using one of the methods describes above. Factors that you should consider when determining the policy frequency include storage availability, backup and log routines, and the rate at which Qlik Replicate processes events.



*If the **Exclusively use sp_repldone within a single task** option is also selected, setting a very short interval (e.g. 1) may adversely affect task processing, and possibly even result in an error.*

- **Alternate backup folder:** The location of the backup logs when using a third-party utility to back up the transaction logs (i.e. instead of Microsoft SQL Server's own backup mechanism). Note that the backup files must be exported to the specified location in standard Microsoft SQL Server format.
- **Change processing mode:** Choose one of the following change processing modes:
 - **Prioritize Online Logs** - This is the default. Replicate will first look for the changes in the online transaction logs. If the changes cannot be found in the online transaction logs, it will look for them in the backup transaction logs instead.
 - **Prioritize Backup Logs** - When this option is enabled, Replicate will first look for the changes in the backup transaction logs. This can improve performance when reading from the online transaction log is slow (e.g due to lock contention) or when using file-level access to access the backup transaction logs.
If the changes cannot be found in the backup transaction logs, it will look for them in the online transaction logs instead.
 - **Backup Logs Only** - When this option is selected, Qlik Replicate will try and find the changes in the backup transaction logs only. Selecting this method results in increased latency due to the interval between backups. The actual latency time will remain constant, but will vary according to the backup schedule.
 - **Online Logs Only** - When this option is selected, Qlik Replicate will try and find the changes in the online transaction logs only.
- **Replicate has file-level access to the backup log files:** Select this option if Qlik Replicate has been granted file-level access to the backup log files in the **Alternate backup folder**.



When Qlik Replicate has file-level access to the backup transaction logs, the following rules apply:

- The **Alternate backup folder** must be a common shared network folder, for example: `\\temp\backup`.
- The **Qlik Replicate Server** service must be configured to log on using the user name and password specified in the **Backup folder user name** and **Backup folder password** fields.

To do this:

In the Windows **Services** console, double-click the **Qlik Replicate Server** service.

In the **Log On** tab, select **This account** and then enter the user name and password.

- The specified user must be granted Read permission to the alternate backup folder (i.e. the shared network folder).

For a complete list of the limitations affecting file-level access, see *Limitation and considerations* (page 940).

- **Backup folder user name:** The user name required to access the backup folder when Qlik Replicate has file-level access.
- **Backup folder password:** The password required to access the backup folder when Qlik Replicate has file-level access.
- **Select virtual backup device types:** When this option is selected, Qlik Replicate will read changes from the specified virtual device(s). Usually, this option only needs to be enabled when using a third-party backup utility (which will be recorded as a virtual device).

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

Working with Windows authentication

You can configure the Qlik Replicate Microsoft SQL Server endpoint to log in to Microsoft SQL Server (on Windows) using Windows authentication.

If you choose this option, you also need to make sure that:

- The Microsoft SQL Server instance is set up to allow Windows log on.
 - The Qlik Replicate user is specified as the "Log on as" user for the "Qlik Replicate Server" service account.
- OR-
- Microsoft SQL Server is configured to allow login for the Qlik Replicate Server service account.

Artifacts created by Replicate in the database

In order to capture changes, Qlik Replicate creates various artifacts in the **dbo** schema of the Microsoft SQL Server database when the task starts. You can safely remove them when the task completes.

The following artifact is created to track open transactions:

- #attrep_dbcc_open_trans_table

The following functions are created to verify that the LSN (Log Sequence Number) is valid:

- attrep_fn_LsnSegmentToHexa
- attrep_fn_NumericLsnToHexa

9.19 Using Microsoft Azure Database for MySQL as a source

This section describes how to set up and use Microsoft Azure Database for MySQL as a source in a replication task.

In this section:

- *Prerequisites (page 369)*
- *Limitations and considerations (page 369)*
- *Permissions (page 370)*
- *MySQL database source data types (page 371)*
- *Setting general connection properties (page 375)*
- *Selecting a schema (page 376)*
- *Setting advanced connection properties (page 377)*

Prerequisites

To use Microsoft Azure Database for MySQL as an endpoint in a Qlik Replicate task, make sure the prerequisites outlined below have been met.

Driver prerequisites

Replicate on Windows

Install MySQL ODBC Unicode Driver 64-bit version 8.0.32 on the Replicate machine.

Replicate on Linux

1. Stop the Replicate service:
`./areplicate stop`
2. Optionally, confirm that the service has stopped:
`./areplicate status`
3. Install MySQL ODBC Unicode Driver 64-bit version 8.0.32 on the Replicate machine.
4. Change the working directory to `<Replicate-Install-Dir>/bin`.
5. Copy the driver location to the `site_arep_login.sh` file:
`echo "export LD_LIBRARY_PATH=\$LD_LIBRARY_PATH:/usr/lib64" >> site_arep_login.sh`
This will add the driver to "LD_LIBRARY_PATH" and update the driver location in the **site_arep_login.sh** file.
6. Optionally, confirm that the driver location was copied:
`cat site_arep_login.sh`
7. Make sure that the `/etc/odbcinst.ini` file contains an entry for MySQL, as in the following example:

```
[MySQL ODBC 8.0 Unicode Driver]
Driver = /usr/lib64/libmyodbc8w.so
UsageCount = 1
```
8. Start the Replicate service:
`./areplicate start`
9. Optionally confirm that the service has started:
`./areplicate status`

Verifying that Microsoft Azure database for MySQL is set up for CDC (Change data capture)

Microsoft Azure Database for MySQL enables binary logging (i.e. CDC) by default. To check that binary logging is indeed enabled, make sure that the Microsoft Azure Database for MySQL **binlog_row_image** parameter is set to **full**.

Replicating 4-byte UTF8 emojis

Replication of 4-byte UTF8 emojis to certain targets requires the following preparation:

- **Microsoft SQL Server Target:** Transform the emojis from `WSTRING (n)` to `WSTRING (n*2)`.
- **Amazon Redshift Target:** Transform the emojis from `WSTRING (n)` to `WSTRING (n*2)`.

For information on defining transformations, see *Using the Transform tab (page 1086)*.

Limitations and considerations

The following limitations apply:

- The following DDLs are not supported:
 - All partition DDLs
 - Drop Table
 - Rename Table
- Changes that are older than one year will not be captured.
- Using the `alter table <table_name> add column <column_name>` statement to add columns to the beginning or to the middle of a table is not supported. When the `alter table <table_name> add column <column_name>` statement is used to add a column to the middle of a table, the column will be added to the end of the table instead.
- The **AR_H_USER** header column is currently not supported. For information on using header columns, see *Headers (page 1143)*.
- If a Microsoft Azure Database for MySQL table contains LOBs and the task's **Replicate Lob columns** option is disabled, the table will be replicated without the LOB columns. Note that this only applies to MEDIUMBLOB, LONGBLOB, MEDIUMTEXT and LONGTEXT columns. This limitation does not apply to BLOB, TINYBLOB, TEXT and TINYTEXT columns.
- If the Microsoft Azure Database for MySQL database is stopped during Full Load, the Full Load will end successfully, but the tables on the target may have less rows than the source tables. If this should happen, either restart the task or reload the tables with the missing rows.
- A Unique Index/Primary Key created on only part of the column data is not supported. If such a Unique Index/Primary Key is encountered, it will be created on the entire target column. The following is an example of a statement that creates a Unique Index/Primary Key using only part of the column data:

```
CREATE INDEX partial_name ON customer (name(10));
```
- Replication of computed columns is supported in Full Load replication tasks only. During CDC, INSERTs to computed columns will appear as NULL on the target and UPDATEs will be ignored.
- Replicate supports zero dates only if both the source and target MySQL-based databases are configured with the same "Strict Mode" (On or Off). If this is not the case, zero dates can be transformed to non-zero dates [using the transformation feature](#).
- The Transaction ID of XA transactions will not be displayed in the replication ChangeTable/Audit Table **transaction_id** header.
- Binary log transaction compression is not supported.
- The MySQL primary-primary configuration is not supported.
- Failover of cluster nodes is not supported.
- Capturing changes from an encrypted binlog is not supported.
- Due to an issue with MySQL ODBC Driver 8.0.027 or later, empty TEXT columns may not be replicated correctly during Full Load. For example, if one table row contains a TEXT column with a value and the same column in the next row contains an empty value (but not NULL), both rows will display the value of the first row on the target.

Permissions

The user specified in the endpoint connection settings must have the following permissions:

Full Load tasks

The ReplicationAdmin role with the SELECT privilege for the designated source tables.

Apply Changes tasks

The ReplicationAdmin role with the REPLICATION CLIENT and REPLICATION SLAVE privileges.



When capturing changes from Microsoft Azure Database for MySQL 8.0, the user must also have Standard authentication.

Checking for open XA transactions

When the task is initialized, Replicate checks whether there are any open XA transactions (transactions that use a two phase commit protocol). To allow this query to be performed, the following permission needs to be granted:

```
grant XA_RECOVER_ADMIN on *.* to 'userName'@'%';
```

Where `userName` is the user specified in the endpoint settings.

If this permission is not granted, the following error will be shown when you run the task:

```
Failed to check for open XA transactions. Verify that the user specified in the endpoint settings has the XA_RECOVER_ADMIN permission. Capturing changes while there are open XA transactions might lead to loss of data. If there are no XA transactions in this database, add the internal parameter IgnoreOpenXaTransactionsCheck to skip the XA transactions check.
```

MySQL database source data types

The following table shows the Microsoft Azure Database for MySQL source data types that are supported when using Qlik Replicate and the default mapping to Qlik Replicate data types. When replicating to a MySQL target, the source and target data types are the same, apart from the exceptions described in *Homogeneous Replication* (page 374).




For information on how to view the data type that is mapped in the target, see the section for the target database you are using.

For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).

Supported data types

Microsoft Azure Database for MySQL source data types	Qlik Replicate Data types
INT	INT4
BIGINT	INT8
MEDIUMINT	INT4
TINYINT	INT1
SMALLINT	INT2
UNSIGNED TINYINT	UINT1
UNSIGNED SMALLINT	UINT2
UNSIGNED MEDIUMINT	UINT4

9 Adding and managing source endpoints

Microsoft Azure Database for MySQL source data types	Qlik Replicate Data types
UNSIGNED INT	UINT4
UNSIGNED BIGINT	UINT8
DECIMAL (10)	NUMERIC (10,0)
BINARY	BYTES (1)
BIT	BOOLEAN
BIT (64)	BYTES (8)
BLOB	BYTES (65535)
LONGBLOB	BLOB
MEDIUMBLOB	BLOB
TINYBLOB	BYTES (255)
DATE	DATE
DATETIME	DATETIME
<div style="border: 1px solid gray; padding: 5px; margin-bottom: 5px;">  <i>DATETIME without a parenthetical value is replicated without milliseconds, whereas DATETIME with a value of 1-5 - e.g. DATETIME(5) - is replicated with milliseconds.</i> </div> <div style="border: 1px solid gray; padding: 5px;">  <i>When replicating a DATETIME column, the time remains the same on the target (i.e. it is not converted to UTC).</i> </div>	
TIME	STRING
TIMESTAMP	DATETIME
<div style="border: 1px solid gray; padding: 5px;">  <i>When replicating a TIMESTAMP column, the time is converted to UTC on the target.</i> </div>	
YEAR	INT2
DOUBLE	REAL8

9 Adding and managing source endpoints

Microsoft Azure Database for MySQL source data types	Qlik Replicate Data types
<p>FLOAT</p> <p>If the FLOAT values are not in the range specified below, use a transformation to map FLOAT to STRING. For an explanation of how to do this, see <i>Using the Transform tab (page 1086)</i>.</p> <p>Supported FLOAT range:</p> <p>- 1.79E+308 to -2.23E-308, 0</p> <p>and</p> <p>2.23E-308 to 1.79E+308</p>	REAL (DOUBLE)
*VARCHAR (45)	WSTRING (45)
*VARCHAR (2000)	WSTRING (2000)
*VARCHAR (4000)	WSTRING (4000)
VARBINARY (4000)	BYTES (4000)
VARBINARY (2000)	BYTES (2000)
*CHAR	WSTRING
*TEXT	WSTRING (65535)
*LONGTEXT	NCLOB
*MEDIUMTEXT	NCLOB
*TINYTEXT	WSTRING (255)
GEOMETRY	BLOB
POINT	BLOB
LINestring	BLOB
POLYGON	BLOB
MULTIPOINT	BLOB
MULTILINestring	BLOB
MULTIPOLYGON	BLOB
GEOMETRYCOLLECTION	BLOB
ENUM	<p>WSTRING (Length)</p> <p>Where "Length" is the longest value in the ENUM.</p>

Microsoft Azure Database for MySQL source data types	Qlik Replicate Data types
SET	WSTRING (Length) Where "Length" is the total of all values in the SET, including commas.
JSON	CLOB



If the DATETIME and TIMESTAMP data types are specified with a “zero” value (i.e. 0000-00-00), you need to make sure that the target database in the replication task supports "zero" values for the DATETIME and TIMESTAMP data types. If they are not supported, you can use a transformation to specify a supported value (e.g. 1970.) Otherwise, they will be recorded as null on the target.

Homogeneous Replication

The following section describes how Replicate handles replication between a MySQL source and the following targets: MySQL, Microsoft Azure Database for MySQL, and Google Cloud SQL for MySQL.



- *In homogeneous replication, the source data first passes through the Qlik Replicate data type and is therefore subject to any limitations of that type.
For information on Replicate data types and their limitations (where relevant), see Replicate data types (page 81).
For information on which Replicate data types the source data passes through when replicating from MySQL, see the MySQL to Qlik Replicate data types mapping table described earlier.*
- *When replicating a TIMESTAMP column, the time is converted to UTC on the target.*

Limitations and considerations

Homogeneous replication is not supported with the following Replicate configurations or operations:

- Apply Changes only tasks
- Stopping and resuming a task
- Starting a task from a timestamp
- When the **If target table already exists** in the task settings' **Full Load Settings** tab is set to **Do nothing**

Data types

When replicating to a MySQL-based target endpoint, the data types will be identical with the following exceptions:

MySQL source and MySQL-based target data types

MySQL source data types	MySQL-based target data types
JSON	LONGTEXT



When using data types from other database engines, the data type on the target will be the corresponding MySQL data type. For more information on using non-native data types, refer to the MySQL online Help.

Collation

Table and column collations will be replicated to the target. Collatable data types are indicated by an asterisk (*) in the *Supported data types (page 371)* table.

To support collation replication, the DBA must ensure that the collations defined for the source MySQL database are the same as those defined for the target MySQL-based database.

Non-nullable columns and primary/unique index names

Non-nullable columns and Primary/Unique Index names are preserved during homogeneous replication.

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a Microsoft Azure Database for MySQL source endpoint to Qlik Replicate:

1. In the Qlik Replicate Console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.
2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the database being used.
3. In the **Description** field, optionally type a description that helps to identify the Microsoft Azure Database for MySQL endpoint.
4. Select **Source** as the endpoint **role**.
5. From the **Type** drop-down list, select **Microsoft Azure Database for MySQL**.



When this endpoint is used as a duplicated source in a Log Stream Staging setup, select the **Read changes from log stream** check box and then select the relevant Log Stream Staging task from the drop-down list.
For information on setting up and managing Log Stream Staging tasks, see *Using the Log Stream (page 1072)*.

6. In the **Server Name** field, enter the host name or IP address of the computer on which the Microsoft Azure Database for MySQL database is installed.
7. Optionally, change the default port (3306).

8. Type the Microsoft Azure Database for MySQL authentication information (**User Name**, **Password**) for the authorized user for this Microsoft Azure Database for MySQL database. If you do not know this information, see your Microsoft Azure Database for MySQL database administrator (DBA).



Consider the following:

- This information is required. If you are using the **Advanced** tab to create a custom string, make sure to include the **User Name** and **Password** properties. See *Setting advanced connection properties* (page 377) for more information.
- This information is case sensitive.
- If you want to set custom properties for this database, see *Setting advanced connection properties* (page 377).



Make sure that the Microsoft Azure Database for MySQL user entered in the Microsoft Azure Database for MySQL Authentication section has the correct access privileges. For information on how to provide the required privileges, see *Permissions* (page 370).

SSL authentication options

In the **Security** section, you can configure Replicate to connect to the Microsoft Azure Database for MySQL database using SSL.

- **SSL Mode:** Select one of the following:
 - **Preferred** - Establishes an encrypted connection if the server supports encrypted connections, falling back to an unencrypted connection if an encrypted connection cannot be established.
 - **None** - Connect with a surname and password only.
 - **Required** - Establishes an encrypted connection if the server supports encrypted connections. The connection attempt fails if an encrypted connection cannot be established.
 - **Verify CA** - Similar to **Required**, but also verifies the server Certificate Authority (CA) certificate against the configured CA certificates. The connection attempt fails if no valid matching CA certificates are found.
 - **Verify Full** - Similar to **Verify CA**, but also performs host name identity verification by checking the host name the client (i.e. Replicate) uses for connecting to the server against the identity in the certificate that the server sends to the client. The client checks whether the host name that it uses for connecting matches the Common Name value in the server certificate. The connection fails if there is a mismatch.
- **Client certificate path** - The path to the client certificate requested by the server.
- **Private key path** - The path to the client private key file in PEM format.
- **CA path** - The path to the Certificate Authority (CA) that issued the client certificate file in PEM format.

Selecting a schema

You can choose which Microsoft Azure Database for MySQL database to access. After configuring the Microsoft Azure Database for MySQL source database connection settings, open the **Select Tables** dialog box (by clicking the **Table Selection** button on the right of the console) and select which schema to use from the

Schema drop down list.

See also *Defining and managing tasks* (page 109).

Setting advanced connection properties

In the **Advanced** tab, you can set the following parameters:

- **Check binary log for new events every:** Specify how often to check the binary log for changes when the endpoints is idle.
- **Additional ODBC connection properties:** Specify any additional ODBC connection parameters that may be required.
- **Database timezone:** Select the timezone where the source database is located.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

9.20 Using a MongoDB database as a source

This section describes how to use a MongoDB database as a source endpoint in a Qlik Replicate task.

In this section:

- *Overview (page 379)*
- *Prerequisites (page 379)*
- *Limitations and considerations (page 380)*
- *Setting general connection properties (page 380)*
- *Setting advanced connection properties (page 383)*

Overview

The following topic provides an overview of using MongoDB as a source endpoint in a Replicate task.

Terminology

The following describes common database terms and their MongoDB equivalents:

Common database terms

Relational Database Term	Corresponding Term in MongoDB
Schema	Database
Table	Collection
Record	Document

With most Replicate source endpoints, when you define a replication task, you can select the tables you wish to replicate from one or more *schemas*. With MongoDB however, instead of selecting schemas, you select databases, and instead of selecting tables, you select collections. So, from a MongoDB perspective, any label in the UI that refers to source tables (e.g. **Table Settings**) is actually referring to source collections, and any label that refers to source records is actually referring to source documents. Likewise, any label in the UI that refers to source schemas is actually referring to source databases.

How changes are handled

Replicate reads the MongoDB oplog for changes (see *Prerequisites (page 379)*). If a change is detected, the modified JSON document will be replicated in its entirety to the target, where it will replace the current JSON document. If the "Store Changes" replication option is also enabled, a record of all operations (UPDATE, DELETE, etc.) will be stored for each JSON document in the associated Change Table(s).

How target tables are created

For each MongoDB collection, a corresponding target table is created with two columns:

- **_id** - a WSTRING column containing the document's `_id` field (the primary key).
- **_doc** - an NCLOB column containing the JSON representation of the document.

Prerequisites

Before you can use MongoDB as a source endpoint in a Replicate task, make sure that the following prerequisites have been met:

- MongoDB must be configured with "majority" read concern level (which requires `supportsSnapshotReadConcern` and `supportsCommittedReads` to be enabled). Note that the WiredTiger storage engine (the default) is already configured with "majority" read concern.
- Before starting a task with the Full Load and/or Apply Changes replication options enabled, set the [Transaction consistency timeout](#) value in the task settings' **Full Load Tuning** tab to **0**. This is important for preventing the capture of old transactions.
- The user specified in the endpoint settings must be granted the "read" role for databases from which you wish to replicate.

- To capture changes, the user specified in the endpoint settings must have privileges that grant `changeStream` and `find` actions on all non-system collections for all databases in the deployment (for example: "readAnyDatabase" role).
- To enable Replicate to detect changes, MongoDB databases servers must be set up in a replica set configuration. In such a configuration, change operations are recorded in the primary server's oplog, which is periodically accessed by Replicate. Note that although a Replica set usually contains multiple servers (one primary and any number of secondary servers), creating a Replica set with only a primary server - i.e. without secondary servers - is also sufficient, as the changes will still be written to the oplog.

Limitations and considerations

Using MongoDB as a source database in a Qlik Replicate task is subject to the following limitations:

- When storing changes in Change Tables, the before image will not be stored.
- When processing an UPDATE operation on the source, the most up-to-date copy of the document will be replicated to the target. For example, if three UPDATE operations were performed on a document, only the document with the last UPDATE will be replicated to target.
- Unlimited LOB size is not supported.
- Capture of new empty databases and new empty collections is not supported. A new collection must contain at least one document before it can be captured.
- [Table content filters](#) are not supported.
- Database names must be English only.
- Dropping or renaming source collections:
 - **During CDC:** If a source collection is renamed/dropped during CDC, it will be suspended (i.e. excluded from the replication).
 - **During Full Load:** If a source collection is renamed/dropped during Full Load, the task will stop.
- Dropping or renaming source databases is not supported.
- Replication of transactional audit data is not supported.
- Replication of a document in a collection that contains a reference to other documents using DBRefs is not supported.

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a MongoDB source endpoint to Qlik Replicate:

1. In the Qlik Replicate Console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.
2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the database being used.

9 Adding and managing source endpoints

3. In the **Description** field, type a description that helps to identify the MongoDB database. This is optional.
4. Select **Source** as the endpoint **role**.
5. From the **Type** drop-down list, select **MongoDB**.



*When this endpoint is used as a duplicated source in a Log Stream Staging setup, select the **Read changes from log stream** check box and then select the relevant Log Stream Staging task from the drop-down list.*

For information on setting up and managing Log Stream Staging tasks, see [Using the Log Stream](#) (page 1072).

6. Select **Standard** or **MongoDB Atlas** for **MongoDB deployment**.
7. For **Standard** deployment, configure the settings according to the table below:

Setting configurations

Option	Description
Hosts	<p>When working with a MongoDB cluster, you can specify multiple hosts in the following format:</p> <pre>host1[:port1][, ...hostN[:portN]]</pre> <p>Replicate will try to connect to the first host. If it is unable to do so, it will then try to connect to the other hosts in the order they are listed.</p>

9 Adding and managing source endpoints

Option	Description
Authentication method	<p>Select one of the following:</p> <ul style="list-style-type: none"> • SCRAM-SHA-256 • SCRAM-SHA-1 <p>When one of the SCRAM authentication methods is selected, you also need to provide the following information:</p> <ul style="list-style-type: none"> • User name: Specify the user name of a user with the required role. • Password: Specify the password of the above user. • Authentication database name: The name of the authentication database on which the specified user account is defined. • x.509 (Certificate) - To make this available for selection, first select the Use SSL option. <p>When this authentication method is selected, you also need to provide the following information:</p> <ul style="list-style-type: none"> • Certificate subject: The subject of your public key certificate. • Keystore file (in JKS format): The full path of the keystore file (containing your private key) on the Replicate Server machine. • Keystore password: The passphrase for accessing your keystore file. • PLAIN (LDAP SASL) <p>When this authentication methods is selected, you also need to provide the following information:</p> <ul style="list-style-type: none"> • User name: Specify the user name of a user with the required role. • Password: Specify the password of the above user. <ul style="list-style-type: none"> • None
Use SSL	<p>Select to transfer data over an encrypted connection.</p> <p>When this option is selected, you also need to provide the following information:</p> <ul style="list-style-type: none"> • Truststore file (in JKS format): The full path of the truststore file on the Replicate Server machine. The trustore file should contain the CA certificate for verifying the MongoDB servers' certificates. • Truststore Password: The password for accessing your truststore.

For **MongoDB Atlas** deployment, configure the settings according to the table below:

Setting configurations

Option	Description
Atlas hostname	<p>When working with MongoDB Atlas, you must specify the host in the following format:</p> <pre>server.example.com</pre>

Option	Description
Authentication method	<p>Select one of the following:</p> <ul style="list-style-type: none"> SCRAM-SHA-1 When this authentication method is selected, you also need to provide the following information: <ul style="list-style-type: none"> User name: Specify the user name of a user with the required role. Password: Specify the password of the above user. Authentication database name: The name of the authentication database on which the specified user account is defined. x.509 (Certificate) - To make this available for selection, first select the Use SSL option. When this authentication method is selected, you also need to provide the following information: <ul style="list-style-type: none"> Certificate subject: The subject of your public key certificate. Keystore file (in JKS format): The full path of the keystore file (containing your private key) on the Replicate Server machine. Keystore password: The passphrase for accessing your keystore file. AWS IAM With this authentication method, you use an IAM credentials source with one of the following options: <ul style="list-style-type: none"> Key pair: Specify the Access key, the Secret key, and the Session token (optional). Environment variables: Must be set by the user. PLAIN (LDAP SASL) When this authentication method is selected, you also need to provide the following information: <ul style="list-style-type: none"> User name: Specify the user name of a user with the required role. Password: Specify the password of the above user.

Setting advanced connection properties

In the **Advanced** tab, you can set the following parameters:

- Set target `_id` column size according to auto-generated ObjectID:** This is the default. It assumes that the source document `_id` has been auto-generated by MongoDB. As an auto-generated ObjectID cannot exceed 24 bytes, the target column `_id` size will be set to 24 bytes. Clear this option if you defined custom IDs that exceed 24 bytes.
- Maximum size of target `_id` column:** If you defined custom IDs that exceed 24 bytes, specify the size of the largest ID. This will enable Replicate to create the `_id` column with a suitable VARCHAR length on the target.

- **Additional connection properties:** Specify any additional connection parameters that may be required. Properties should be specified in the following format:
key1=value1&key2=value2...

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

9.21 Using a MySQL-based database as a source

This section describes how to set up and use a MySQL-based database as a source in a replication task.

You need to configure the Replicate MySQL endpoint when replicating from any of the following databases:

- MySQL
- Percona
- MariaDB
- Microsoft Azure Database for MySQL - Flexible Server
- Amazon Aurora



The procedures for configuring connectivity to these endpoints are identical to those described in this section, for MySQL. However, when using Percona as a source, the CDC prerequisites described in Prerequisites (page 386) are not relevant.

In this section:

- *Prerequisites (page 386)*
- *Limitations and considerations (page 388)*
- *Required permissions (page 389)*
- *MySQL database source data types (page 390)*
- *Setting general connection properties (page 394)*
- *Selecting a schema (page 395)*
- *Setting advanced connection properties (page 395)*

Prerequisites

The following section describes the prerequisites for working with the Qlik Replicate MySQL endpoint.

Supported versions and editions

For information on supported MySQL versions, see [the ReplicateSupport Matrix](#).

The following MySQL editions are supported:

- MySQL Community Edition
- MySQL Standard Edition
- MySQL Enterprise Edition
- MySQL Cluster Carrier Grade Edition

Driver prerequisites

Replicate on Windows

Install MySQL ODBC Unicode Driver 64-bit version 8.0.32 on the Replicate machine.

Replicate on Linux

1. Stop the Replicate service:

```
./areplicate stop
```
2. Optionally, confirm that the service has stopped:

```
./areplicate status
```
3. Install MySQL ODBC Unicode Driver 64-bit version 8.0.32 on the Replicate machine.
4. Change the working directory to `<Replicate-Install-Dir>/bin`.
5. Copy the driver location to the `site_arep_login.sh` file:

```
echo "export LD_LIBRARY_PATH=\$LD_LIBRARY_PATH:/usr/lib64" >> site_arep_login.sh
```

This will add the driver to "LD_LIBRARY_PATH" and update the driver location in the **site_arep_login.sh** file.
6. Optionally, confirm that the driver location was copied:

```
cat site_arep_login.sh
```
7. Make sure that the `/etc/odbcinst.ini` file contains an entry for MySQL, as in the following example:

```
[MySQL ODBC 8.0 Unicode Driver]
Driver = /usr/lib64/libmyodbc8w.so
UsageCount = 1
```
8. Start the Replicate service:

```
./areplicate start
```
9. Optionally confirm that the service has started:

```
./areplicate status
```

General prerequisites

Make sure the following prerequisites have been met:

- A MySQL account with the *Required permissions* (page 389).
- MySQL 8: Standard authentication for capturing changes

9 Adding and managing source endpoints

- Certified MySQL versions support the fractional value for temporal data types (introduced in MySQL 5.6). Consequently, replication of the old temporal data type format is no longer supported.



Note that upgrading to MySQL versions that support the fractional value for temporal data types does not upgrade the temporal data type format automatically. This needs to be done separately. This also applies when upgrading from MariaDB 10.1. For instructions on upgrading the temporal data type format, see the vendor's online help.

Enable binary logging

To enable binary logging (required for CDC replication), the following parameters must be configured in MySQL's `my.ini` (Windows) or `my.cnf` (UNIX) files.

Binary logging parameters

Parameter	Value
<code>server_id</code>	Any value from 1. Example: <code>server_id=1</code>
<code>log-bin=<path></code>	Path to the binary log file (without an extension). Example: <code>log-bin=E:\MySQL_Logs\BinLog</code>
<code>binlog_format</code>	Must be: <code>binlog_format=row</code>
<code>expire_logs_days</code>	To prevent disk space issues, it is strongly recommended not to use the default value (0). Example: <code>expire_logs_days=5</code>
<code>binlog_row_image</code>	Must be: <code>binlog_row_image=full</code>

Cluster prerequisites

To be able to replicate clustered (NDB) tables (by connecting to any of the cluster nodes), the following parameters must be configured in MySQL's `my.ini` (Windows) or `my.cnf` (UNIX) files.



- *Replication from Galera Cluster for MySQL/MariaDB is also supported. For more information, see **Galera Cluster Considerations** below.*
- *When using Percona as a source, there is no need to perform the procedures described in this section.*

Cluster parameters

Parameter	Value
ndb_log_bin	<p>Must be:</p> <p>ndb_log_bin=on</p> <p>Ensures that changes in clustered tables will be logged to the binary log.</p>
ndb_log_update_as_write	<p>Must be:</p> <p>ndb_log_update_as_write=OFF</p> <p>Prevents UPDATES being written as INSERTs in the binary log.</p>
ndb_log_updated_only	<p>Must be:</p> <p>ndb_log_updated_only=OFF</p> <p>Ensures that the binary log will contain the entire row and not just the changed columns.</p>

Galera cluster considerations

You can replicate data from any active MySQL instance. In the event that an instance to which Replicate is currently connected fails, Replicate will attempt to reconnect several times. If it is unable to reconnect, the task will fail. In this case, you will need to manually change the server instance (by editing the endpoint settings) and then start the task from the timestamp of the first failure. If the exact point of failure is unknown, it can be retrieved from the log. Make sure to specify the timestamp of the first failure (i.e. before Replicate started to try to reconnect) or even a few minutes earlier.

Replicating 4-byte UTF8 emojis

Replication of 4-byte UTF8 emojis to certain targets requires the following preparation:

- **Microsoft SQL Server Target:** Transform the emojis from `WSTRING(n)` to `WSTRING(n*2)`.
- **Amazon Redshift Target:** Transform the emojis from `WSTRING(n)` to `WSTRING(n*2)`.

For information on defining transformations, see *Using the Transform tab (page 1086)*.

Limitations and considerations

You should be aware of the following limitations and considerations when using a MySQL source:

- When MySQL is installed on Windows, changes are not captured from tables whose names contain both upper and lower case characters.

- If the MySQL database is stopped during Full Load, the Full Load will end successfully, but the tables on the target might have less rows than the source tables. If this should happen, either restart the task or reload the tables with the missing rows.
- A Unique Index/Primary Key created on only part of the column data is not supported. If such a Unique Index/Primary Key is encountered, it will be created on the entire target column. The following is an example of a statement that creates a Unique Index/Primary Key using only part of the column data:

```
CREATE INDEX partial_name ON customer (name(10));
```
- Changes to computed (virtual) columns will not be captured.
- Binary log transaction compression is not supported.
- The MySQL primary-primary configuration is not supported.
- Failover of cluster nodes is not supported.
- Capturing changes from an encrypted binlog is not supported.
- Due to an issue with MySQL ODBC Driver 8.0.027 or later, empty TEXT columns may not be replicated correctly during Full Load. For example, if one table row contains a TEXT column with a value and the same column in the next row contains an empty value (but not NULL), both rows will display the value of the first row on the target.
- Capture of the following DDLs is not supported:
 - All partition DDLs
 - Drop Table
 - Rename Table
- Changes that are older than one year will not be captured.
- Using the `alter table <table_name> add column <column_name>` statement to add columns to the beginning or to the middle of a table is not supported. When the `alter table <table_name> add column <column_name>` statement is used to add a column to the middle of a table, the column will be added to the end of the table instead.
- Zero dates are only supported if both the source and target MySQL-based databases are configured with the same "Strict Mode" (On or Off). If this is not the case, zero dates can be transformed to non-zero dates using [the transformation feature](#).
- The Transaction ID of XA transactions will not be displayed in the replication ChangeTable/Audit Table **transaction_id** header.
- The **AR_H_USER** header column is currently not supported. For more information on header columns, see *Headers* (page 1143).
- If a MySQL table contains LOBs and the task's **Replicate Lob columns** option is disabled, the table will be replicated without the LOB columns. Note that this only applies to MEDIUMBLOB, LONGBLOB, MEDIUMTEXT and LONGTEXT columns. This limitation does not apply to BLOB, TINYBLOB, TEXT and TINYTEXT columns.

Required permissions

The user specified in the endpoint connection settings must have the following permissions:

Full Load tasks

The ReplicationAdmin role with the SELECT privilege for the designated source tables.

Apply Changes tasks

The ReplicationAdmin role with the REPLICATION CLIENT and REPLICATION SLAVE privileges.

Checking for open XA transactions

When the task is initialized, Replicate checks whether there are any open XA transactions (transactions that use a two phase commit protocol). To allow this query to be performed, the following permission needs to be granted:

```
grant XA_RECOVER_ADMIN on *.* to 'userName'@'%';
```

Where `userName` is the user specified in the endpoint settings.

If this permission is not granted, the following error will be shown when you run the task:

```
Failed to check for open XA transactions. Verify that the user specified in the endpoint settings has the XA_RECOVER_ADMIN permission. Capturing changes while there are open XA transactions might lead to loss of data. If there are no XA transactions in this database, add the internal parameter IgnoreOpenXaTransactionsCheck to skip the XA transactions check.
```

MySQL database source data types

The following table shows the MySQL database source data types that are supported when using Qlik Replicate and the default mapping to Qlik Replicate data types. When replicating to a MySQL target, the source and target data types are the same, apart from the exceptions described in *Homogeneous replication* (page 393).




For information on how to view the data type that is mapped in the target, see the section for the target database you are using.

For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).

Supported data types

MySQL Source Data Types	Qlik Replicate Data Types
INT	INT4
BIGINT	INT8
MEDIUMINT	INT4
TINYINT	INT1
SMALLINT	INT2
UNSIGNED TINYINT	UINT1
UNSIGNED SMALLINT	UINT2
UNSIGNED MEDIUMINT	UINT4
UNSIGNED INT	UINT4
UNSIGNED BIGINT	UINT8
DECIMAL (10)	NUMERIC (10,0)

9 Adding and managing source endpoints

MySQL Source Data Types	Qlik Replicate Data Types
BINARY	BYTES (1)
BIT	BOOLEAN
BIT (64)	BYTES (8)
BLOB	BYTES (65535)
LONGBLOB	BLOB
MEDIUMBLOB	BLOB
TINYBLOB	BYTES (255)
DATE	DATE
DATETIME	DATETIME
<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;">  <i>DATETIME without a parenthetical value is replicated without milliseconds, whereas DATETIME with a value of 1-5 - e.g. DATETIME(5) - is replicated with milliseconds.</i> </div> <div style="border: 1px solid #ccc; padding: 5px;">  <i>When replicating a DATETIME column, the time remains the same on the target (i.e. it is not converted to UTC).</i> </div>	
TIME	STRING
TIMESTAMP	DATETIME
<div style="border: 1px solid #ccc; padding: 5px;">  <i>When replicating a TIMESTAMP column, the time is converted to UTC on the target.</i> </div>	
YEAR	INT2
DOUBLE	REAL8
FLOAT	REAL (DOUBLE)
<p>If the FLOAT values are not in the range specified below, use a transformation to map FLOAT to STRING. For an explanation of how to do this, see <i>Using the Transform tab (page 1086)</i>.</p> <p>Supported FLOAT range:</p> <p>- 1.79E+308 to -2.23E-308, 0</p> <p>and</p> <p>2.23E-308 to 1.79E+308</p>	

9 Adding and managing source endpoints

MySQL Source Data Types	Qlik Replicate Data Types
*VARCHAR (45)	WSTRING (45)
*VARCHAR (2000)	WSTRING (2000)
*VARCHAR (4000)	WSTRING (4000)
VARBINARY (4000)	BYTES (4000)
VARBINARY (2000)	BYTES (2000)
*CHAR	WSTRING
*TEXT	WSTRING (65535)
*LONGTEXT	NCLOB
*MEDIUMTEXT	NCLOB
*TINYTEXT	WSTRING (255)
GEOMETRY	BLOB
POINT	BLOB
LINESTRING	BLOB
POLYGON	BLOB
MULTIPOINT	BLOB
MULTILINESTRING	BLOB
MULTIPOLYGON	BLOB
GEOMETRYCOLLECTION	BLOB
ENUM	WSTRING (Length) Where "Length" is the longest value in the ENUM.
SET	WSTRING (Length) Where "Length" is the total of all values in the SET, including commas.
JSON	CLOB



If the DATETIME and TIMESTAMP data types are specified with a “zero” value (i.e. 0000-00-00), you need to make sure that the target database in the replication task supports “zero” values for the DATETIME and TIMESTAMP data types. If they are not supported, you can use a transformation to specify a supported value (e.g. 1970.) Otherwise, they will be recorded as null on the target.

Homogeneous replication

The following section describes how Replicate handles replication between a MySQL source and the following targets: MySQL, Microsoft Azure Database for MySQL, and Google Cloud SQL for MySQL.



- *In homogeneous replication, the source data first passes through the Qlik Replicate data type and is therefore subject to any limitations of that type. For information on Replicate data types and their limitations (where relevant), see Replicate data types (page 81). For information on which Replicate data types the source data passes through when replicating from MySQL, see the MySQL to Qlik Replicate data types mapping table described earlier.*
- *When replicating a `TIMESTAMP` column, the time is converted to UTC on the target.*

Limitations and considerations

Homogeneous replication is not supported with the following Replicate configurations or operations:

- Apply Changes only tasks
- Stopping and resuming a task
- Starting a task from a timestamp
- When the **If target table already exists** in the task settings' **Full Load Settings** tab is set to **Do nothing**

Data types

The data types will be identical with the following exceptions:

Exceptions	
MySQL Source Data Types	MySQL-based Target Data Types
JSON	LONGTEXT



When using data types from other database engines, the data type on the target will be the corresponding MySQL data type. For more information on using non-native data types, refer to the MySQL online Help.

Collation

Table and column collations will be replicated to the target. Collatable data types are indicated by an asterisk (*) in the *Supported data types (page 390)* table.

To support collation replication, the DBA must ensure that the collations defined for the source MySQL database are the same as those defined for the MySQL-based target database.

Non-nullable columns and primary/unique index names

Non-nullable columns and Primary/Unique Index names are preserved during homogeneous replication.

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.



You can also use MySQL files as a source. For more information, see [Using the Qlik Replicate file channel \(page 1061\)](#).

To add a MySQL source endpoint to Qlik Replicate:

1. In the Qlik Replicate Console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see [Defining and managing endpoints \(page 114\)](#).
2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the MySQL database. This is optional.
4. Select **Source** as the endpoint **role**.
5. From the **Type** drop-down list, select **MySQL**.



*When this endpoint is used as a duplicated source in a Log Stream Staging setup, select the **Read changes from log stream** check box and then select the relevant Log Stream Staging task from the drop-down list.*

For information on setting up and managing Log Stream Staging tasks, see [Using the Log Stream \(page 1072\)](#).

6. In the **Server Name** field, enter the host name or IP address of the computer on which the MySQL database is installed.
7. Optionally, change the default port (3306).
8. Type the MySQL authentication information (**User Name**, **Password**) for the authorized user for this MySQL database. If you do not know this information, see your MySQL database Administrator (DBA).



Consider the following:

- *This information is required. If you are using the **Advanced** tab to create a custom string, make sure to include the **User Name** and **Password** properties. See [Setting advanced connection properties \(page 395\)](#) for more information.*
- *This information is case sensitive.*
- *If you want to set custom properties for this database, see [Setting advanced connection properties \(page 395\)](#).*



Make sure that the MySQL user entered in the MySQL Authentication section has the correct access privileges. For information on how to provide the required privileges, see [Required permissions \(page 389\)](#).

SSL authentication options

In the **Security** section, you can configure Replicate to connect to the MySQL database using SSL.

- **SSL Mode:** Select one of the following:
 - **Preferred** - Establishes an encrypted connection if the server supports encrypted connections, falling back to an unencrypted connection if an encrypted connection cannot be established.
 - **None** - Connect with a surname and password only.
 - **Required** - Establishes an encrypted connection if the server supports encrypted connections. The connection attempt fails if an encrypted connection cannot be established.
 - **Verify CA** - Similar to **Required**, but also verifies the server Certificate Authority (CA) certificate against the configured CA certificates. The connection attempt fails if no valid matching CA certificates are found.
 - **Verify Full** - Similar to **Verify CA**, but also performs host name identity verification by checking the host name the client (i.e. Replicate) uses for connecting to the server against the identity in the certificate that the server sends to the client. The client checks whether the host name that it uses for connecting matches the Common Name value in the server certificate. The connection fails if there is a mismatch.
- **Client certificate path** - The path to the client certificate requested by the server.
- **Private key path** - The path to the client private key file in PEM format.
- **CA path** - The path to the Certificate Authority (CA) that issued the client certificate file in PEM format.

Selecting a schema

You can choose which MySQL database to access. After configuring the MySQL source database connection settings, open the **Select Tables** dialog box (by clicking the **Table Selection** button on the right of the console) and select which schema to use from the **Schema** drop down list.

See also *Defining and managing tasks* (page 109).

Setting advanced connection properties

In the **Advanced** tab, you can set the following parameters:

- **Check binary log for new events every:** How often to check the binary log for changes when the database is idle (i.e. when no changes are done to the source tables). The default is five seconds. Usually there is no reason to change the interval. However, to reduce the load on the system, you might want to increase the interval in the following scenarios:
 - The database is updated infrequently
 - Polling the system for changes slows down the system
 - Changes are not required in real time
- **Additional ODBC connection properties:** Specify any additional ODBC connection parameters that may be required.
- **Database timezone:** Select the timezone where the source database is located.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

9.22 Using ODBC to connect to a source

This section describes how to use ODBC connectivity to connect to a source endpoint.



The ODBC source endpoint can be used to access sources that are not included in Qlik Replicate's extensive Support Matrix. However, unless enabled through Professional Services and approved by your Account Manager, the ODBC source endpoint should not be used. Additionally, the ODBC source endpoint should not be used to access any sources already listed in the Support Matrix.

In this section:

- *Prerequisites (page 398)*
- *Limitations and considerations (page 398)*
- *ODBC source data types (page 399)*
- *Setting general connection properties (page 402)*
- *Setting advanced connection properties (page 404)*

Prerequisites

The following section describes the prerequisites for working with Qlik Replicate and an ODBC source endpoint.

Qlik Replicate Server for Windows

You can connect an endpoint to Qlik Replicate using ODBC by indicating the DSN (Data Source Name). In this case you must be sure that a DSN is defined for the ODBC endpoint on the computer where Qlik Replicate is installed.

1. Install an endpoint client on the computer where Qlik Replicate is installed. The client you install depends on the ODBC provider you are using. For example, if you are using an IBM DB2 endpoint, install an IBM DB2 client.



You must use a 64-bit ODBC provider client to work with Qlik Replicate.

2. Use the ODBC Data Source Administrator to create a System DSN. The Data Source is located in the Windows control panel.

Qlik Replicate Server for Linux

The following section describes the steps you need to perform to work with Qlik Replicate for Linux and ODBC as a source endpoint in a Qlik Replicate task.

1. On the Qlik Replicate Server machine, install the ODBC client that you want to use (e.g. postgresQL).
2. Make sure that the `/etc/odbcinst.ini` file contains the correct entry for the driver you installed, as in the following example:

```
[PostgreSQL]
Description = ODBC for PostgreSQL
Driver = /usr/lib/psqlodbc.so
Setup = /usr/lib/libodbcpsqlS.so
Driver64 = /usr/lib64/psqlodbc.so
Setup64 = /usr/lib64/libodbcpsqlS.so
FileUsage = 1
```



To access an IBM DB2 for LUW target using ODBC, make sure that you specify the `libdb2o.so` driver (and not `libdb2.so`).

3. Define a DSN for the installed driver by editing the `/etc/odbc.ini` file, as in the following example:

```
[Postgre_DSN]
Description = Test
Driver = /usr/lib64/psqlodbc.so
Endpoint = MyDatabase
Servername = 12.3.45.678
Port = 5432
```

Limitations and considerations

When using ODBC as a source, the following limitations apply:

9 Adding and managing source endpoints

- UPDATES to primary key fields are not supported. To update the field, define it as a unique index instead.
- The ODBC Source endpoint supports full-load operations only.
- For providers that do not support batch operations, you must set the `RowByRow=true` internal parameter according to the description provided in [Internal Parameters](#).
- Replicating data with 4-byte emoji characters is not supported.

ODBC source data types

The following table shows the ODBC source data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.

For information on how to view the data type that is mapped in the target, see the section for the target endpoint you are using.




For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.


Supported data types

ODBC Data Types	Qlik Replicate Data Types
SQL_BIT	BOOLEAN
SQL_TINYINT	INT1 UINT1 SQL data types are mapped to unsigned data types when the <code>UNSIGNED_ATTRIBUTE</code> is set to <code>SQL_TRUE</code> for the data type being mapped.
SQL_SMALLINT	INT2 UINT2 SQL data types are mapped to unsigned data types when the <code>UNSIGNED_ATTRIBUTE</code> is set to <code>SQL_TRUE</code> for the data type being mapped.
SQL_INTEGER	INT4 UINT4 SQL data types are mapped to unsigned data types when the <code>UNSIGNED_ATTRIBUTE</code> is set to <code>SQL_TRUE</code> for the data type being mapped.

9 Adding and managing source endpoints

ODBC Data Types	Qlik Replicate Data Types
SQL_BIGINT	INT8 UINT8 SQL data types are mapped to unsigned data types when the <code>UNSIGNED_ATTRIBUTE</code> is set to <code>SQL_TRUE</code> for the data type being mapped.
SQL_DOUBLE	REAL8
SQL_FLOAT	REAL8
SQL_REAL	REAL8
SQL_NUMERIC (P,S)	NUMERIC (P,S) REAL8 The <code>SQL_NUMERIC</code> data type is mapped to <code>REAL8</code> when <i>at least one</i> of the following is true: <ul style="list-style-type: none"> • Precision > 38 • Scale < 0 • Scale > 38 • Scale > Precision
SQL_DECIMAL (P,S)	NUMERIC (P,S) REAL 8 The <code>SQL_NUMERIC</code> data type is mapped to <code>REAL8</code> when <i>at least one</i> of the following is true: <ul style="list-style-type: none"> • Precision > 38 • Scale < 0 • Scale > 38 • Scale > Precision
SQL_DATE	DATE
SQL_TYPE_DATE	
SQL_TIME	TIME
SQL_TYPE_TIME	
SQL_TIMESTAMP	DATETIME
SQL_TYPE_TIMESTAMP	

ODBC Data Types	Qlik Replicate Data Types
SQL_CHAR SQL_VARCHAR	STRING
SQL_WCHAR SQL_WVARCHAR	WSTRING
SQL_LONGVARCHAR <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> To use this data type with Qlik Replicate, you must enable the use of CLOBs for a specific task.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p> </div>	CLOB
SQL_WLONGVARCHAR <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> To use this data type with Qlik Replicate, you must enable the use of NCLOBs for a specific task.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p> </div>	NCLOB
SQL_BINARY	BYTES
SQL_LONGVARBINARY <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> To use this data type with Qlik Replicate, you must enable the use of BLOBs for a specific task.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p> </div>	BLOB
SQL_GUID	STRING

ODBC Data Types	Qlik Replicate Data Types
SQL_INTERVAL_YEAR SQL_INTERVAL_MONTH SQL_INTERVAL_DAY SQL_INTERVAL_MINUTE SQL_INTERVAL_HOUR SQL_INTERVAL_SECOND SQL_INTERVAL_YEAR_TO_MONTH SQL_INTERVAL_DAY_TO_HOUR SQL_INTERVAL_DAY_TO_MINUTE SQL_INTERVAL_DAY_TO_SECOND SQL_INTERVAL_HOUR_TO_MINUTE SQL_INTERVAL_HOUR_TO_SECOND SQL_INTERVAL_MINUTE_TO_SECOND	STRINGw
Provider specific data types <div style="border: 1px solid gray; padding: 10px;">  <p><i>If column length is 0 or > 4000 then:</i></p> <p><i>To use this data type with Qlik Replicate, you must enable the use of BLOBs for a specific task.</i></p> <p><i>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</i></p> </div>	If column length is < or = 4000: BYTES If column length is 0 or > 4000: BLOB

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add an ODBC source endpoint to Qlik Replicate:

1. In the Qlik Replicate Console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button.

2. In the **Name** field, type a name for your ODBC endpoint. This can be any name that will help to identify the endpoint being used.
3. In the **Description** field, type a description that helps to identify the ODBC endpoint. This is optional.
4. Select **Source** as the endpoint **role**.
5. Select **ODBC** as the endpoint **Type**.
6. Select one of the following:
 - If the DSN you want to use is not included in the list, make sure that the endpoint client is installed on the computer with Qlik Replicate and that the DSN is defined. Note that the ODBC provider client must be 64-bit. For more information, see *Prerequisites* (page 398).



*If you are using an ARC CDC Agent as the source in a Qlik Replicate task, you cannot select the DSN for the Qlik ODBC driver as the target. In this case, to use Qlik ODBC as a source, you must enter the connection string manually by selecting **Connection String** and following the directions for that option in this procedure.*

- **Connection String:** Select this to connect to an ODBC-supported endpoint using a connection string then type a valid connection string in the field below. For information on how to create a connection string, see the documentation for the ODBC endpoint provider you are using. Note that if you specify a password in your connection string, it will be revealed as plain text in the task log files. It is therefore recommended to specify the password in the GUI **Password** field.



- You can use the **Advanced** tab to add specific properties and create a custom connect string. In this case, you do not need to enter information in this tab. For more information on using the **Advanced** tab, see *Setting advanced connection properties* (page 404).
- To determine if you are connected to the endpoint you want to use or if the connection information you entered is correct, click **Test Connection**. If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box. To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

7. Type the authentication information (**User Name**, **Password**) for the authorized user for the ODBC endpoint being used. For example, the IBM DB2 system administrator if you are using a IBM DB2 provider. If you do not know this information, see your ODBC Endpoint System Administrator.



- When you select **Connection String** be sure to include **User name/password** information in the connection string that you type in the box. If you are using the **Advanced** tab to create a custom string, make sure to include the **User Name** and **Password** properties. For more information, see *Setting advanced connection properties* (page 404).
- This information is case sensitive.
- You can set custom properties in the **Advanced** tab. For more information, see *Setting advanced connection properties* (page 404).



Make sure that the ODBC endpoint user has the correct access privileges for the ODBC provider being used.

Setting advanced connection properties

In the **Advanced** tab, you can set the following properties:

- **Provider syntax:** Select the name of the provider syntax if you are using an alternate provider syntax.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

9.23 Using ODBC with CDC as a source

This section describes how to use ODBC connectivity to connect to a source endpoint in a Full Load and/or CDC task.



The ODBC with CDC source endpoint can be used to access sources that are not included in Qlik Replicate's extensive Support Matrix. However, unless enabled through Professional Services and approved by your Account Manager, the ODBC with CDC source endpoint should not be used. Additionally, the ODBC with CDC source endpoint should not be used to access any sources already listed in the Support Matrix.

In this section:

- *Prerequisites (page 406)*
- *Limitations and Considerations (page 406)*
- *ODBC with CDC source data types (page 407)*
- *Setting general connection properties (page 410)*
- *Setting Change Processing parameters (page 412)*

Prerequisites

The following section describes the prerequisites for working with Qlik Replicate and an ODBC source with CDC.

Replicate server for Windows

You can connect an endpoint to Qlik Replicate using ODBC by indicating the DSN (Data Source Name). In this case you must be sure that a DSN is defined for the ODBC endpoint on the computer where Qlik Replicate is installed.

1. Install an endpoint client on the computer where Qlik Replicate is installed. The client you install depends on the ODBC provider you are using. For example, if you are using an IBM DB2 endpoint, install an IBM DB2 client.



You must use a 64-bit ODBC provider client to work with Qlik Replicate

2. Use the ODBC Data Source Administrator to create a System DSN. The Data Source is located in the Windows control panel.

Replicate server for Linux

The following section describes the steps for working with Qlik Replicate for Linux and ODBC with CDC as a source endpoint in a Replicate task.

1. On the Qlik Replicate Server machine, install the ODBC client that you want to use (e.g. postgresQL).
2. Make sure that the `/etc/odbcinst.ini` file contains the correct entry for the driver you installed, as in the following example:

```
[PostgreSQL]
Description = ODBC for PostgreSQL
Driver = /usr/lib/psqlodbc.so
Setup = /usr/lib/libodbcpsqlS.so
Driver64 = /usr/lib64/psqlodbc.so
Setup64 = /usr/lib64/libodbcpsqlS.so
FileUsage = 1
```

3. Define a DSN for the installed driver by editing the `/etc/odbc.ini` file, as in the following example:

```
[Postgre_DSN]
Description = Test
Driver = /usr/lib64/psqlodbc.so
Endpoint = MyDatabase
Servername = 12.3.45.678
Port = 5432
```

Limitations and Considerations

The following limitations apply:

- UPDATES to primary key fields are not supported. To update the field, define it as a unique index instead.
- For providers that do not support batch operations, you must manually add the `RowByRow=true` internal parameter according to the instruction provided in *Setting Change Processing parameters* (page 412).

- The [Start from timestamp](#) run option is supported only if there is a single context column defined in the [Change Processing tab](#), and its type is `TIMESTAMP`.
- Replicating data with 4-byte emoji characters is not supported.

ODBC with CDC source data types

The following table shows the ODBC target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.

For information on how to view the data type that is mapped in the target, see the section for the target endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).

Supported data types

ODBC Data Types	Qlik Replicate Data Types
SQL_BIT	BOOLEAN
SQL_TINYINT	INT1 UINT1 SQL data types are mapped to unsigned data types when the <code>UNSIGNED_ATTRIBUTE</code> is set to <code>SQL_TRUE</code> for the data type being mapped.
SQL_SMALLINT	INT2 UINT2 SQL data types are mapped to unsigned data types when the <code>UNSIGNED_ATTRIBUTE</code> is set to <code>SQL_TRUE</code> for the data type being mapped.
SQL_INTEGER	INT4 UINT4 SQL data types are mapped to unsigned data types when the <code>UNSIGNED_ATTRIBUTE</code> is set to <code>SQL_TRUE</code> for the data type being mapped.
SQL_BIGINT	INT8 UINT8 SQL data types are mapped to unsigned data types when the <code>UNSIGNED_ATTRIBUTE</code> is set to <code>SQL_TRUE</code> for the data type being mapped.
SQL_DOUBLE	REAL8

9 Adding and managing source endpoints

ODBC Data Types	Qlik Replicate Data Types
SQL_FLOAT	REAL8
SQL_REAL	REAL8
SQL_NUMERIC (P,S)	<p>NUMERIC (P,S)</p> <p>REAL8</p> <p>The SQL_NUMERIC data type is mapped to REAL8 when <i>at least one</i> of the following is true:</p> <ul style="list-style-type: none"> • Precision > 38 • Scale < 0 • Scale > 38 • Scale > Precision
SQL_DECIMAL (P,S)	<p>NUMERIC (P,S)</p> <p>REAL 8</p> <p>The SQL_NUMERIC data type is mapped to REAL8 when <i>at least one</i> of the following is true:</p> <ul style="list-style-type: none"> • Precision > 38 • Scale < 0 • Scale > 38 • Scale > Precision
SQL_DATE	DATE
SQL_TYPE_DATE	
SQL_TIME	TIME
SQL_TYPE_TIME	
SQL_TIMESTAMP	DATETIME
SQL_TYPE_TIMESTAMP	
SQL_CHAR	STRING
SQL_VARCHAR	
SQL_WCHAR	WSTRING
SQL_WVARCHAR	

ODBC Data Types	Qlik Replicate Data Types
<p>SQL_LONGVARCHAR</p> <p>To use this data type with Qlik Replicate, you must enable the use of CLOBs for a specific task.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p>	CLOB
<p>SQL_WLONGVARCHAR</p> <p>To use this data type with Qlik Replicate, you must enable the use of NCLOBs for a specific task.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p> <p>For more information, see LOB support in Task Settings/Metadata (page 1166).</p>	NCLOB
SQL_BINARY	BYTES
<p>SQL_LONGVARBINARY</p> <p>To use this data type with Qlik Replicate, you must enable the use of BLOBs for a specific task.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p> <p>For more information, see LOB support in Task Settings/Metadata (page 1166).</p>	BLOB
SQL_GUID	STRING

ODBC Data Types	Qlik Replicate Data Types
SQL_INTERVAL_YEAR SQL_INTERVAL_MONTH SQL_INTERVAL_DAY SQL_INTERVAL_MINUTE SQL_INTERVAL_HOUR SQL_INTERVAL_SECOND SQL_INTERVAL_YEAR_TO_MONTH SQL_INTERVAL_DAY_TO_HOUR SQL_INTERVAL_DAY_TO_MINUTE SQL_INTERVAL_DAY_TO_SECOND SQL_INTERVAL_HOUR_TO_MINUTE SQL_INTERVAL_HOUR_TO_SECOND SQL_INTERVAL_MINUTE_TO_SECOND	STRING
Provider specific data types If column length is 0 or > 4000 then: To use this data type with Qlik Replicate, you must enable the use of BLOBs for a specific task. During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.	If column length is < or = 4000: BYTES If column length is 0 or > 4000: BLOB


Setting general connection properties

This section describes how to configure general connection properties.


To add an ODBC with CDC source endpoint to Qlik Replicate:

1. In the Qlik Replicate console, click the Manage Endpoint Connections toolbar button to open the Manage Endpoints Connections dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.
2. In the **Name** field, type a name for your ODBC endpoint. This can be any name that will help to identify the endpoint being used.
3. In the **Description** field, type a description that helps to identify the ODBC endpoint. This is optional.


4. Select **Source** as the endpoint **role**.
5. Select **ODBC with CDC** as the endpoint **Type**.

 When this endpoint is used as a duplicated source in a Log Stream Staging setup, select the **Read changes from log stream** check box and then select the relevant Log Stream Staging task from the drop-down list.
For information on setting up and managing Log Stream Staging tasks, see *Using the Log Stream* (page 1072).


6. Select one of the following:
 - **DSN:** Select this to connect to an ODBC-supported endpoint using a DSN. When you select DSN you must select the DSN you are using from the list.
If the DSN you want to use is not included in the list, make sure that the endpoint client is installed on the computer with Qlik Replicate and that the DSN is defined. Note that the ODBC provider client must be 64-bit. For more information, see *Prerequisites* (page 406).

 If you are using a Replicate Connect CDC Agent as the source in a Replicate task, you cannot select the DSN for the Qlik ODBC driver as the target. In this case, to use Qlik ODBC as a source, you must enter the connection string manually by selecting **Connection String** and following the directions for that option in this procedure.

- **Connection String:** Select this to connect to an ODBC-supported endpoint using a connection string then type a valid connection string in the field below. For information on how to create a connection string, see the documentation for the ODBC endpoint provider you are using. Note that if you specify a password in your connection string, it will be revealed as plain text in the task log files. It is therefore recommended to specify the password in the GUI **Password** field.

 To determine if you are connected to the endpoint you want to use or if the connection information you entered is correct, click **Test Connection**.
If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.
To view the log entry if the connection fails, click **View Log** (this button is not available unless the test connection fails). The server log is displayed with the information for the connection failure.

7. Type the authentication information (**User Name, Password**) for the authorized user for the ODBC endpoint being used. For example, the IBM DB2 system administrator if you are using a IBM DB2 provider. If you do not know this information, see your ODBC Endpoint System Administrator.

 Consider the following:

- If you select **Connection String**, be sure to include **User name/password** information in the connection string that you type in the box.
- This information is case sensitive.



Make sure that the ODBC endpoint user has the correct access privileges for the ODBC provider being used.

Setting Change Processing parameters

The **Change Processing** tab lets you define change processing settings for the source database. Normally, Replicate scans a database's transaction logs for changes and then applies those changes to the target database. However, this method of change processing is not possible with Data Warehouse endpoint types since these endpoints do not generate transaction logs.

The good news is that you can still use Replicate to capture changes from the database - it just requires a little bit of preparation.

Prerequisites

Before you can define the settings in the **Change Processing** tab, you need to ensure that at least one special "Context" column exists in your source database tables. Context column(s) are basically columns in a table that enable Replicate to determine whether the data has changed. You can add Context columns specifically for the purpose of change processing (either using a script or manually) or you can use existing columns that contain suitable "Context" data.



You can create and reference any number of Context columns in a table as long as the Context column names are the same for all source tables. Additionally, each value in the Context column(s) must be unique.



When working with non-datetime Context columns such as integers that are not based on epoch time (for example, counters, strings, and numeric values), the values of those columns should be created by a singleton entity. This is required to ensure that the results are returned in the correct sequence (thereby preventing loss of data) when the Context column values are generated by multiple concurrent processes. It is strongly recommended not to generate Context column values from multiple concurrent sources as doing so poses a high risk of losing events.

In the example below, the Context column **cf** has been added to the table. The **cf** column contains TIMESTAMPS that enable Replicate to determine whether a change occurred (by comparing the current TIMESTAMP with the TIMESTAMP stored in its repository).

By default, all changes are assumed to be INSERTs. If UPDATE and DELETE operations are also performed on the source tables, you can write an UPDATE and/or DELETE expression (described below) that will enable Replicate to identify the operation type.

a	first_name	cf ▼	oper
1	may1	2014-12-29 14:52:2...	D
2	may2	2014-12-29 14:52:2...	D
6	omri6	2014-12-29 14:53:5...	D
8	omri8	2014-12-29 14:53:5...	D
9	omri9	2014-12-29 14:53:5...	U
10	omri10	2014-12-29 14:53:5...	U
11	omri11	2014-12-29 14:53:5...	U
3	pitz	2014-12-30 11:04:0...	U
7	gal	2014-12-30 13:27:1...	U
5	gal	2014-12-30 13:29:2...	U
12	gal	2014-12-30 13:41:5...	U

Limitations

The following limitations apply when Change Processing is enabled:

- The "Start from timestamp" run option is not supported. For more information, see *Advanced Run Options (page 1205)*.
- If one of the Context columns is part of the Primary Key or Unique Index, then UPDATE and DELETE operations are not supported.
- Context columns cannot be LOB columns
- DDLs are not supported
- When inserting a record and then updating the same record, the task error handling settings should be set as follows:
 1. Open the **<Task Name> Settings** dialog box.
 2. Select the **Error Handling|Apply Conflicts** tab.
 3. Set a task-specific Apply Conflicts policy as described in *Error Handling settings (page 1187)*.
 4. From the **No record found for applying an update** drop-down list, select **INSERT the missing target record**.

For more information on error handling, see *Error handling (page 1186)*.

Configuring Change Processing settings

Perform the following steps to configure change processing settings.

To configure change processing settings:

1. Select the endpoint's **Change Processing** tab.
2. In the **Columns** field, specify the names of the Context columns. The column names are case-sensitive and must be separated by commas.
Example:
context1,context2
3. Choose the sorting order of the Context columns as appropriate (**Ascending** or **Descending**). Note that if the order you select is not the same as the actual sorting order, an error will occur.
4. In the **Check for changes every** field, specify how often to check for changes.
5. Enter expressions that Replicate will use to identify UPDATE and DELETE operations. If you do not enter any expressions or if no match is found for an expression, any row whose context is higher (if the sorting order is **Ascending**) or lower (if the sorting order is **Descending**) than the previous context value will be considered an INSERT.
Expressions must be written in the native syntax of the source database. All examples in this section are written using PostgreSQL syntax.
 - **Update expression** - Enter an expression for identifying UPDATE operations.
Example (based on [Figure "Example of a Table with a Context Column"](#)):

```
case when oper='U' then 1 else 0 end
```

Selecting the **UPDATE the existing target record** option in the *Apply Conflicts (page 1191)* tab, eliminates the need to provide an UPDATE expression.
 - **Delete expression** - Enter an expression for identifying DELETE operations.
Example (based on [Figure "Example of a Table with a Context Column"](#)):

```
case when oper='D' then 1 else 0 end
```

In addition to the DELETE expression, DELETE operations should be carried out as "Soft" deletes. This means that the row is not actually deleted from the table, but rather, marked as "deleted".
6. Select **Override connection string parameters** to append the connection string with parameters that are not exposed in the UI. As such parameters are normally not required, they should only be used after consulting with Qlik Support.

Additional settings and summary

In the **Change Processing** tab, you can also set internal parameters, set more options, and view a summary of your settings.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

9.24 Using Oracle as a source

This section describes how to set up and use an Oracle database as a source in a replication task.

In this section:

- *Supported Oracle database editions (page 417)*
- *Prerequisites (page 417)*
- *Required permissions (page 420)*
- *Limitations and considerations (page 423)*
- *Supported encryption methods (page 426)*
- *Supported compression methods (page 427)*
- *Supported standby configurations (page 428)*
- *Redo log files - access method guidelines (page 428)*
- *Handling shrink space operations (page 431)*
- *Replicating nested tables (page 431)*
- *Oracle source data types (page 432)*
- *Homogeneous Replication (page 436)*
- *Working with Oracle on Oracle Cloud (page 437)*
- *Working with Amazon RDS for Oracle (page 437)*
- *Setting general connection properties (page 440)*
- *Setting advanced connection properties (page 442)*
- *Finding the wallet entries used for TDE Encryption (page 450)*

Supported Oracle database editions

Replicate supports the following Oracle database editions:

- Oracle Enterprise Edition
- Oracle Standard Edition
- Oracle Standard Edition 2
- Oracle Express Edition



From Oracle 18.x, change capture from Oracle Express Edition is not supported.

- Oracle Personal Edition

Prerequisites

The following section describes the prerequisites for working with the Qlik Replicate Oracle endpoint.

Install the required driver



Support for the XMLTYPE data type requires the full Oracle Client.

Replicate On Windows

Install Oracle Instant Client for Microsoft Windows (x64) Version 12.0 or later.

Replicate On Linux

1. Stop the Replicate service:
`./areplicate stop`
2. Optionally, confirm that the service has stopped:
`./areplicate status`
3. Install Oracle Instant Client for Linux (x86-64) Version 12.0 or later.
4. In addition, if not already included in your system, create a symbolic link in the `$oracle_home/lib` directory. Name the link `libclntsh.so`, and point it to a specific version of this file.

Example - Oracle 12c client:

```
lrwxrwxrwx 1 oracle oracle 63 Oct 2 14:16 libclntsh.so ->
/u01/app/oracle/home/lib/libclntsh.so.12.1
```

5. Change the working directory to `<Replicate-Install-Dir>/bin`.and do one of the following according to your driver version:

Oracle Instant Client versions 12-18:

Append the `LD_LIBRARY_PATH` environment variable to the Oracle `lib` directory by copying the driver location to the `site_arep_login.sh` file:

```
echo "export LD_LIBRARY_PATH=\$LD_LIBRARY_
PATH:/u01/app/oracle/home/lib/" > site_arep_login.sh
```

Oracle Instant Client version 19 or later:

Append the LD_LIBRARY_PATH environment variable to the Oracle instant client directory and to the Replicate lib directory by copying the driver location to the site_arep_login.sh file:

```
echo "export LD_LIBRARY_PATH=\$LD_LIBRARY_
PATH:/opt/oracle/instantclient_<version>/opt/attunity/replicate/lib/" >
site_arep_login.sh
```

6. Optionally, confirm that the driver location was copied:

```
cat site_arep_login.sh
```

7. Start the Replicate service:

```
./areplicate stop
```

8. Optionally, confirm that the service has started:

```
./areplicate status
```

Provide Oracle account access

You must provide Oracle account access to the Qlik Replicate user. This user must have read/write privileges on the Oracle database. For information on setting up access to the Oracle account, see *Required permissions* (page 420).

Ensure that ARCHIVELOG mode is on

Oracle can be run in two different modes: the ARCHIVELOG mode and the NOARCHIVELOG mode. To use the Oracle logs with Qlik Replicate, run the database in ARCHIVELOG mode. If the log is not set to ARCHIVELOG mode, then execute the following query:

```
ALTER database ARCHIVELOG
```

Note that if your Oracle database instance is on Amazon RDS, a different command needs to be executed. For more information, see *Enabling automatic backups* (page 439) and *Setting up archiving* (page 439) in *Working with Amazon RDS for Oracle* (page 437).

Set up supplemental logging

Supplemental logging must be enabled for the Oracle database.



- You can automatically set up supplemental logging in the **Advanced** tab of the Oracle database dialog box. If you select this option, you do not have to carry out the following procedure. For more information, see *Setting advanced connection properties using Oracle LogMiner* (page 442) and *Setting advanced connection properties using Replicate Log Reader* (page 444) according to your redo logs access method.
- When replicating from an Oracle standby environment, supplemental logging must be set up in the primary Oracle instance (as described below), not in the standby instance.

Set up supplemental logging as described in the steps below.

Step 1: Check that supplemental logging is enabled for the database

1. Run the following query:

```
SELECT name, value, description FROM v$parameter WHERE name = 'compatible';
```

The returned result should be from GE to n.n.n where n.n.n is the Oracle database version (e.g. 10.0.0).



For Replicate to work, the parameter value must match the real version of the database.

2. Run the following query:

```
SELECT supplemental_log_data_min FROM v$database;
```

The returned result should be YES or IMPLICIT.
Enable supplemental logging by executing the following query:

```
ALTER DATABASE ADD SUPPLEMENTAL LOG DATA
```



If your Oracle database instance is on Amazon RDS, a different command needs to be executed. For more information, see [Working with Amazon RDS for Oracle](#) (page 437).

Step 2: Make sure that the required supplemental logging is added for all columns on each table

1. If a Primary Key exists, supplemental logging must be added for the Primary Key either by using the format to add supplemental logging on the Primary Key, or by adding supplemental logging on the Primary Key columns.
2. If no Primary Key exists and the table has a single Unique Index, then all of the Unique Index's columns must be added to the supplemental log. Using `SUPPLEMENTAL LOG DATA (UNIQUE INDEX) COLUMNS` does not add the Unique Index columns to the log.
3. If no Primary Key exists and the table has multiple Unique Indexes, Qlik Replicate will select the first Unique Index. Qlik Replicate will use the first index in an alphabetically ordered ascending list. Supplemental logging must be added on the selected index's columns. Using `SUPPLEMENTAL LOG DATA (UNIQUE INDEX) COLUMNS` does not add the Unique Index columns to the log.
4. If there is no Primary Key and no Unique Index, supplemental logging must be added on all columns.



When the target table Primary Key/Unique Index is different than the source table Primary Key/Unique Index, the user needs to add supplemental logging manually on the source table columns that comprise the target table Primary Key/Unique Index.

5. If you change the target table primary key, the supplemental logging must be added on the selected index's columns instead of the columns of the original primary key/unique index.

Step 3: If a filter or transformation is defined for the table, additional logging might be necessary



If ALL COLUMNS supplemental logging has been added to the table, there is no need to add any additional logging.

If the table has a Unique Index or a Primary Key, you also need to add supplemental logging on each column that is involved in a filter or transformation (if those columns are different than the Primary Key or Unique Index columns).



If a transformation uses only one column, this column may not be added to a supplemental logging group. For example, "A+B" needs both columns to be added, whereas substring(A, 10) does not need "A" to be added.

One method of setting up both Primary Key/Unique Index supplemental logging and supplemental logging on specific columns is to add USER_LOG_GROUP supplemental logging only on the Primary Key/Unique Index columns and on the columns that are filtered or transformed.

For example, to replicate a table named EXAMPLE.TABLE with Primary Key ID and filter by column NAME, you can run a command similar to the one below to create the log group supplemental logging:

```
ALTER TABLE EXAMPLE.TABLE ADD SUPPLEMENTAL LOG GROUP example_log_group (ID,NAME) ALWAYS;
```

Step 4: When the [Insert the missing target record](#) Apply Conflicts option is selected, supplemental logging must be enabled for ALL the source table columns.

Required permissions

This section describes the permissions that are required for working with an Oracle source.



When replicating from Oracle for Amazon RDS, see [Required Amazon RDS for Oracle permissions](#) (page 437) instead.

General permissions

To use an Oracle source in a Replicate task, the user specified in the Replicate Oracle endpoint connection settings must be granted the following privileges in the Oracle database:



If any of the required privileges cannot be granted to a V\$xxx, then grant them to the V_\$xxx.

- SELECT ANY TRANSACTION
- SELECT on V_\$ARCHIVED_LOG
- SELECT on V_\$LOG
- SELECT on V_\$LOGFILE
- SELECT on V_\$DATABASE
- SELECT on V_\$THREAD
- SELECT on V_\$PARAMETER
- SELECT on V_\$NLS_PARAMETERS
- SELECT on V_\$TIMEZONE_NAMES

- SELECT on GV_\$TRANSACTION
- SELECT on V_\$CONTAINERS
- SELECT on ALL_INDEXES
- SELECT on ALL_OBJECTS
- SELECT on DBA_OBJECTS - Required if the Oracle version is earlier than 11.2.0.3.
- SELECT on ALL_TABLES
- SELECT on ALL_USERS
- SELECT on ALL_CATALOG
- SELECT on ALL_CONSTRAINTS
- SELECT on ALL_CONS_COLUMNS
- SELECT on ALL_TAB_COLS
- SELECT on ALL_IND_COLUMNS
- SELECT on ALL_LOG_GROUPS
- SELECT on SYS.DBA_REGISTRY
- SELECT on SYS.OBJ\$
- SELECT on SYS.ENC\$
- SELECT on DBA_TABLESPACES
- SELECT on ALL_TAB_PARTITIONS
- SELECT on ALL_ENCRYPTED_COLUMNS
- If views are exposed: SELECT on ALL_VIEWS

Grant the following additional privilege (for each source table) when you are using a specific table list:

- SELECT on <any-source-table>;

Grant the following additional privilege when using a pattern for the table list:

- SELECT ANY TABLE;

Grant the following additional privilege (for each source table) when Replicate adds supplemental logging automatically (the default behavior) and you are using a specific table list. For information on how to turn off supplemental logging, see *Setting advanced connection properties using Oracle LogMiner (page 442)*.

- ALTER on <any-source-table>;

Grant the following additional privilege when Replicate adds supplemental logging automatically (the default behavior). For information on how to turn off supplemental logging, see *Setting advanced connection properties using Oracle LogMiner (page 442)*.

- ALTER ANY TABLE;
- When accessing an Oracle standby database, the following privilege must be granted:
SELECT on V\$STANDBY_LOG
- When accessing nested tables, the following privileges must be granted:
SELECT on all_nested_tables
SELECT on all_nested_table_cols

Access privileges when using Oracle LogMiner to access the redo logs

If you are using Oracle LogMiner to access the Redo logs, grant the following privileges.

- CREATE SESSION
- EXECUTE on DBMS_LOGMNR
- SELECT on V_\$LOGMNR_LOGS
- SELECT on V_\$LOGMNR_CONTENTS
- LOGMINING



This privilege is only required for Oracle 12c or later.

Access privileges when using Replicate Log Reader to access the redo logs

When using Replicate Log Reader to access the Redo logs, the following privileges should be granted to the user specified in the Oracle endpoint settings:

- CREATE SESSION
- SELECT on V_\$TRANSPORTABLE_PLATFORM
Grant the SELECT on v_\$transportable_platform privilege if the Redo logs are stored in ASM and accessed by Replicate from ASM.
- SELECT ON V_\$DATABASE_INCARNATION
- CREATE ANY DIRECTORY
Qlik Replicate uses following Oracle file access features:
- BFILE read - Used when Replicate does not have file-level access to the Redo logs, and the Redo logs are not accessed from ASM.
- DBMS_FILE_TRANSFER package - Used to copy the Redo log files to a temporary folder (in which case the EXECUTE ON DBMS_FILE_TRANSFER privilege needs to be granted as well)
- DBMS_FILE_GROUP package - Used to delete the Redo log files from a temporary/alternate folder (in which case the EXECUTE ON DBMS_FILE_GROUP privilege needs to be granted as well).

Oracle file features work together with Oracle directories. Each Oracle directory object includes the name of the folder containing the files which need to be processed.

If you want Replicate to create and manage the Oracle directories, you need to grant the CREATE ANY DIRECTORY privilege specified above. Note that the directory names will be prefixed with `attrep_`. If you do not grant this privilege, you need to create the corresponding directories manually. The names of directories that are created manually must not be prefixed with `ATTUREP_`, `ATTUTMP_`, or `ATTUGRP_`. If you create the directories manually and the Oracle user specified in the Oracle Source endpoint is not the user that created the Oracle Directories, grant the READ on DIRECTORY privilege as well.

If the Oracle source endpoint is configured to copy the Redo log files to a temporary folder, and the Oracle user specified in the Oracle source endpoint is not the user that created the Oracle directories, the following additional privileges are required:

- READ on the Oracle directory object specified as the source directory
- WRITE on the directory object specified as the destination directory in the copy process

See also: *Setting advanced connection properties (page 442)*.

Required ASM privileges

The following section describes the additional permissions that are required when the redo logs are stored in ASM.

Grant the following read privilege:

```
SELECT ON v_$transportable_platform
```

From Oracle 11g Release 2 (11.2.0.2), Replicate must be granted the SYSASM privilege in order to access the ASM account. For older supported versions, granting Replicate the SYSDBA privilege should be sufficient.



When connecting to ASM, Replicate will first try to log in as SYSDBA and, if unsuccessful, will try to log in as SYSASM.

You can validate ASM account access by opening a command prompt and issuing the following statements:

```
sqlplus asmuser/asmpassword@+asmserver as sysdba
```

-OR-

```
sqlplus asmuser/asmpassword@+asmserver as sysasm
```

Limitations and considerations

You should be aware of the following limitations and considerations when using an Oracle source.

- When replicating from a CHAR column, any trailing spaces will be truncated.
- Function-based indexes are not supported.
- If you are managing supplemental logging and you carry out transformations on any of the columns, you must be sure that supplemental logging is activated for all fields and columns.
- Capturing changes from a pluggable Oracle database (PDB) requires the source endpoint to connect to that PDB instance. Capturing changes from multiple PDB instances by connecting to the container database (CDB) is not supported.
- Deferred Constraints are not supported.
- Changes made by the Oracle DBMS_REDEFINITION package -(e.g. table metadata and the OBJECT_ID) - will not be captured by Qlik Replicate.
- During Change Processing, columns without supplemental logging (e.g. LOB columns) that are not updated will be inserted as NULL in the Change Table.
- During Change Processing, batch updates to numeric columns defined as a Primary Key are not supported.


Example of an unsupported UPDATE command:

```
UPDATE tableX set ID=ID+1;
```

Where tableX is the table name and ID is a numeric column defined as a Primary Key.

- LONG and LONG RAW column limitations:

- Data in LONG and LONG RAW columns cannot exceed 64k. Any data that exceeds 64k will be truncated.
- The **Allow unlimited LOB size** option is not supported when loading LONG and LONG RAW columns into Oracle target.
Workaround: Use the [Limit LOB size to \(KB\)](#) option instead.
- Tables whose names contain apostrophes cannot be replicated.
- Change Data Capture (CDC) is not supported from dynamic views.
- Index-organized tables with an overflow segment are not supported in Change Processing (CDC) tasks.
- Changes to Index-organized tables with key compression are not supported.
- Replication of tables or views of type BIN\$ or DR\$ is not supported.
- Oracle Flashback Database and restore points are not supported as the related operations affect the consistency of Oracle redo log files.
- When using the escape character '\ ' to search for a table in the Select Tables window, you need to write '\\ ' (2 backslashes) instead of '\ ' (1 backslash). For example, to search for all tables that include '_', you would need to enter %_%.


 - *When adding a pattern for include or exclude, 2 backslashes are not required (for example, *%_%).*
 - *When using '\$' for the escape character, there is no limitation.*

- When using Oracle 19.0 as the source, the following Oracle features are not supported:
 - Data-guard DML redirect
 - Partitioned hybrid tables
 - Schema-only Oracle accounts
- When replicating from Oracle 21c, the following limitations apply:
 - Accessing the redo logs with LogMiner is not supported. Only Replicate Log Reader is supported.
 - Immutable tables are not supported
 - Blockchain tables are not supported
- Object names exceeding 30 characters are not supported. Consequently, tables with names exceeding 30 characters or tables containing column names exceeding 30 characters will not be replicated.
- Replication from Application Containers is not supported.
- Cascading standby database is not supported.
- Failover configuration is supported only when changes are captured from the primary node.
- The AR_H_USER header column is supported only for Oracle database version 11.2.0.3 and higher. In Oracle database version 10, the value for this column may not be correct. For information on using header columns, see *Headers (page 1143)*.
- The rename table <table name> to <new table name> syntax is supported by Qlik Replicate when using Oracle version 11 and higher.

- Data changes resulting from partition/sub-partition operations (ADD, DROP, EXCHANGE and TRUNCATE) will not be replicated and may cause the following errors:
 - For ADD operations, updates and deletes on the added data may return a "0 rows affected" warning.
 - For DROP and TRUNCATE operations, new inserts may result in "duplicates" errors.
 - For EXCHANGE operations, both a "0 rows affected" warning and "duplicates" errors may be encountered.

To replicate changes resulting from partition/sub-partition operations, you need to reload the tables in question. When adding a new *empty* partition, operations on the newly added partition will be replicated to the target as normal.

- When Limited-size LOB mode is enabled, empty BLOB/CLOB columns on the Oracle source are replicated as NULL values. For more information on Limited-size LOB mode, see *Target metadata (page 1166)*.
- During CDC or during Full Load when the **Allow unlimited LOB size** option is enabled, LOB data types are supported only in tables with a primary key or unique index.

Limitations when using Oracle LogMiner to access the redo logs

- UPDATEs to XMLTYPE and LOB columns are not supported (i.e. replicated).
- SHRINK SPACE operations are not supported
- Capturing changes from a PDB using Oracle LogMiner is not supported. Therefore, if you want to connect to a PDB, make sure that the **Access redo logs via** Replicate Log Reader option is selected in the **Advanced** tab.
- Due to unpredictable behavior, it is not recommended to use Oracle LogMiner when replicating from a RAC environment.
- If you change the original primary key column with a transformation, UPDATEs might not be applied to the target. To prevent this, you need to manually enable supplemental logging for the original primary key column.

Limitations when using Replicate Log Reader to access the redo logs

- Table clusters are not supported.
- Only table-level SHRINK SPACE operations are supported. These include the full table, partitions, and sub-partitions.
- Implementing online redo logs on raw devices is not supported.
- The following Oracle features are not supported when using Amazon RDS for Oracle:
 - ASM
 - TDE
- Direct-load INSERT procedure with the parallel execution option is not supported in the following cases:
 - Uncompressed tables with more than 255 columns
 - Row size exceeds 8K
 - Exadata HCC tables

Unsupported data types

The following data types are not supported:

- BFILE
- ROWID
- REF
- UROWID
- ANYDATA
- SDO_GEOMETRY
- User-defined data types



- *Virtual columns are not supported.*
- *As the ROWID data type is not supported, materialized views based on a ROWID column are also not supported.*

Supported encryption methods

This section describes which encryption methods Qlik Replicate supports when working with an Oracle source database, and details the requirements for HSM encryption support.

TDE encryption

TDE encryption support

Redo Logs Access Method	Tablespace	Column
Replicate Log Reader	Yes	Yes
Oracle LogMiner	Yes	Yes

Limitations and considerations

- NNE (Native Network Encryption) is supported as long as it is configured correctly both on Oracle and in the Oracle client **sqlnet.ora** file.
- Replication of columns encrypted with the 3DES168 algorithm is not supported.

HSM encryption

HSM encryption support

Redo Logs Access Method	Tablespace	Column
Replicate Log Reader	Yes	Yes
Oracle LogMiner	Yes	Yes

Prerequisites

- An Oracle DBA should install the PKCS #11 client on the Replicate machine and configure it to work with HSM.

9 Adding and managing source endpoints

- Define the `AREP_HSM_LIB` environment variable with a path to `pkcs11 dll/so`.

Example:

- `set AREP_HSM_LIB=C:\Program Files\Vormetric\DataSecurityExpert\Agent\pkcs11\bin\vorpkcs11.dll`
- `export AREP_HSM_LIB=/opt/cloudhsm/lib/libcloudhsm_pkcs11.so`

Setup

Once the above prerequisites have been fulfilled, configure the **Secret Store Names** and **Secret Store Values** fields in the Oracle endpoint's **General** tab as follows:

- Enter `HSM` in **Secret Store Names** field.
- In the **Secret Store Values** field, set either `crypto_username:password` or just `password` depending on your HSM. For Oracle Key Vault, the password should be the same as the one used to install the **okvclient.jar**. The password or `crypto_username:password` combination are the same credentials that were used to create the TDE master key.

Limitations and considerations

- When working with Oracle Key Vault for Oracle 19, Oracle 19.17 full client or later must be installed.



Oracle Key Vault can be used with any HSM using the PKCS#11 API. Using Oracle Vault in OCI (Oracle Cloud Infrastructure) is not supported, as it does not provide a public PKCS#11 API.

- Replication of columns encrypted with the 3DES168 algorithm is not supported.

Supported compression methods

The table below lists which compression methods Qlik Replicate supports when working with an Oracle source database. As the table shows, compression support depends both on your Oracle database version and whether or not Qlik Replicate is configured to use Oracle LogMiner to access the redo logs.

Supported compression methods

Version	Basic	OLTP	HCC (from Oracle 11g R2)	Others
Oracle 11 or later - Replicate Log Reader	Yes	Yes	Yes	No
Oracle 11 or later - Oracle LogMiner	Yes	Yes	Yes	Yes (Any compression method supported by Oracle LogMiner)



When the Oracle source endpoint is configured to use Replicate Log Reader, the Query Low level of the HCC compression method is only supported in the **Full Load** task mode.

Supported standby configurations

The following table lists the supported Oracle Standby configurations.

Supported configurations

Standby Configuration	Supported	Notes
Snapshot Standby	Yes	-
Physical Standby	Yes	<ul style="list-style-type: none"> • The physical standby database must have Active Data Guard enabled (available from Oracle 11g or later). • When the Oracle LOG_ARCHIVE_DEST_n initialization parameter is configured with the DELAY option, running a Full Load and Apply Changes task on an Oracle Physical Standby instance may result in missing or erroneous data in the target database. This happens because Replicate interprets the timestamp of changes in the Redo log as the timestamp of the Oracle instance from where it performed the Full Load, whereas it is in fact the timestamp of the Oracle instance <i>where the changes occurred</i> (i.e. the Primary Oracle instance). The resulting timestamp discrepancy means that some of the changes in the Redo log may be interpreted as “old” and not be replicated. This issue can be resolved using an internal parameter. For information on this parameter, please contact Qlik Support.
Logical Standby	No	-
Hot Standby	No	-

Redo log files - access method guidelines

The Oracle source endpoint can be configured to access online and archived Oracle redo log files using either Oracle LogMiner (Oracle’s built-in method) or Replicate Log Reader (Replicate’s high-speed redo log reader).

Generally, it is recommended to use Replicate Log Reader as it is more efficient, faster, and uses less resources.

Replicate Log Reader is especially recommended in the following situations:

- The volume of changes in the redo log is more than 30GB/hour
- The volume of changes is between 10GB/hour and 30GB/hour and the changes need to be processed (i.e. replicated to the target) as fast as possible.
- There are multiple tasks replicating from the same source. Using Oracle LogMiner is less efficient in this case, as it accesses the redo logs via the database, thereby consuming additional database resources.

Both Replicate Log Reader and Oracle LogMiner are subject to certain limitations, support different compression methods, and require different access permissions. It is therefore strongly recommended to review the relevant sections before configuring the endpoint settings. In the end, your decision whether to use Replicate Log Reader or Oracle LogMiner may be based on a limitation that exists in one but not the other, the way your data is compressed, or the permissions that you are willing to grant the Replicate user.

See also:

- *Limitations and considerations (page 423)*
- *Supported compression methods (page 427)*
- *Required permissions (page 420)*

Working with an Oracle Data Guard environment



Data Guard is supported when using Replicate Log Reader to access the redo logs only.


In this topic, you will learn how to support switchover and failover operations when replicating from an Oracle Data Guard environment with multiple nodes consisting of one primary node and several standby nodes.



The Oracle source endpoint can only be configured to access one Oracle server or ASM server. Therefore, you need to set up two Oracle source endpoints in Replicate, one for the primary database and the other for the standby database. It is possible to configure one TNS entry that will always point to the active Oracle server, but there is no equivalent for the ASM connection. In addition, the settings in the standby environment and in the primary environment may be different (for example, a different archive log destination ID).

Update task/endpoint credentials

After a failover or switchover operation in a Data Guard environment, it is necessary to update the task/source endpoint. This can be done in one of the following ways:

- Update the source endpoint credentials by changing them from primary to standby, and vice versa.
- Maintain two Oracle source endpoints, one for primary and one for standby. Following a switchover or failover operation, switch between primary and standby nodes. All relevant tables need to be re-selected after making the switch.
- Use QEM command `ReconfigureEndpointNowait` to switch between primary and standby Oracle source endpoints. This command will switch between the endpoints in a task without requiring the user to re-select the task table.
For more information, see  [Qlik Enterprise Manager](#).
- Create two tasks, one that points to the primary database and one that points to the secondary database. Ensure that the first task is active and that the second is stopped. Following a failover operation, the first task will stop and the second task will start. This procedure can also be scripted.



This option does not apply to switchover operations.

Handle switchover scenario:

When a task runs with a primary node and a switchover is executed on the database side:

1. Stop the task.
2. Switch the endpoint from the primary node to a standby node. If it has a variant name and only the IP address changes, do not change server name.
3. Resume the task. The task will continue from the checkpoint where the task was stopped, and will continue applying changes if there are any.

Handle failover scenario:

Replicate preparation

The following procedure describes how to configure Replicate to handle a failover operation in a Data Guard environment:

1. Create a task that points to the primary database.
2. In the advanced settings for the primary Oracle source endpoint, set the `supportResetLog=true` internal parameter.
For more information, see *Additional settings and summary (page 414)*.
3. Run the task. It will be active until a failover occurs.

Failover procedure

In case of a failover scenario, perform the following steps:

1. Ensure that the task pointing to the primary database has stopped. If it hasn't, stop it.
2. Open a command prompt (on Windows) or Linux shell (on Linux) on the Replicate server, and change directory to the bin folder of the Replicate installation.
The default folder is `C:\Program Files\Attunity\Replicate\bin` for Windows and `/opt/attunity/Replicate/bin` for Linux.
3. Run the following command. It will output a checkpoint string that will be used when starting the task.
`repctl -d <data_folder> getcheckpoint <mytask>`
Where `data_folder` is the location of Replicate's data folder and `mytask` is the task name.

Example:

```
repctl -d "C:\Program Files\Attunity\Replicate\data" getcheckpoint "dg1task"
```

The output will be similar to the following:

```
command getcheckpoint response:
```

```
{
  "checkpoint":
  "checkpoint:v1#608#00000000.02180113.00000001.0000.00.0000:354.8739.16#0#0#*#0#13"
}
```

4. Run the following command to start the task with the checkpoint string from the previous step. This will ensure that the task starts from the correct position, so that no events are lost.
`repctl -d <data_folder> connect; execute <mytask> flags=09 operation=02 cdcposition=<checkpoint>; disconnect;`
Where `<data_folder>` is the location of Replicate's data folder, `<mytask>` is your task name, and `<checkpoint>` is the checkpoint string.

Example:

```
repctl -d "C:\Program Files\Attunity\Replicate\data" connect; execute dg1task flags=09 operation=02
```

```
cdcposition=checkpoint:v1#608#00000000.02180113.00000001.0000.00.0000:354.8739.16#0#0#*  
#0#13; disconnect;
```

5. Replicate will now capture all the events that were not previously captured, and continue capturing future events. Ensure that the failover process has finished before restarting the task from where it was stopped, and that the current primary is set to work again.



If you updated your source endpoint by creating two tasks, one active and one stopped, you can do the failover procedure with both.

For more information, see [Update task/endpoint credentials](#) (page 429).

Handling shrink space operations

When a SHRINK SPACE operation occurs, Replicate will capture all of the changes logged to the redo log as a result of the operation and ignore them.

The following message will appear in the task's log file:

Operations generated by the SHRINK SPACE process were ignored.

Monitoring considerations

When Replicate captures changes resulting from a SHRINK SPACE operation, the task's Incoming Changes bar will indicate an unusually large number of changes. However, these changes will not be reflected in the Applied Changes pie chart or the Applied Changes Details table.

See also the *Limitations and considerations* (page 423) section for the limitations related to SHRINK SPACE operations in Replicate Log Reader and LogMiner mode.

Replicating nested tables

Replicate supports the replication of Oracle tables containing columns that are nested tables or defined types. To enable this functionality, select the **Support nested tables** option in the **Advanced** tab.

Replicate creates the target tables of Oracle nested tables as regular tables without a unique constraint. As you will most likely need join the parent and child tables for meaningful data, it is important to manually create a non-unique index on the **NESTED_TABLE_ID** column in the target child table. The **NESTED_TABLE_ID** column can then be used in the JOIN ON clause, together with the parent column corresponding to the child table name. Additionally, creating such an index will improve performance when the target child table data is updated/deleted by Replicate.

It is recommended to [configure the task to stop after Full Load completes](#). After Full Load completes, manually create non-unique indexes for all the replicated child tables on the target, and then resume the task.

If a captured nested table is added to an existing parent table (captured or not captured), Replicate will handle it correctly, but the non-unique index for the corresponding target table will not be created. Note that in this case, if the target child table becomes extremely large, performance may be impacted. In such a case, it is recommended to stop the task, create the index, and then resume the task.

After the nested tables are replicated to the target, the DBA will need to [run a JOIN statement](#) on the parent and corresponding child tables in order to flatten the data.

Prerequisites

Make sure that you replicate parent tables for all the replicated nested tables. Both the parent tables (the tables containing the nested table column) and the child (i.e. nested) tables will be available for selection in Replicate.

Supported nested table types

The following nested table types are supported:

- Data type
- User defined Object

Limitations

- Only one level of nesting is supported.
- Replicate does not verify that both the parent and child table(s) are selected for replication. In other words, it's possible to select a parent table without a child and vice versa.

How nested tables are replicated

The parent and nested tables are replicated to the target as follows:

- The parent table is created identical to the source. The nested column will be defined as RAW(16) and contain a reference to its nested tables in the **NESTED_TABLE_ID** column.
- The child table is created identical to the source, but with an additional column named **NESTED_TABLE_ID** with the same type as the parent nested column and with the same meaning.

JOIN statement example

To flatten the parent table, the DBA should run a JOIN statement between the parent and child tables, as shown in the following example:

Creating the Type table:

```
CREATE OR REPLACE TYPE my_tab_t AS TABLE OF VARCHAR2(30);
```

Creating the parent table with a column of type my_tab_t that was defined above:

```
CREATE TABLE my_parent_table (id NUMBER PRIMARY KEY, col1 my_tab_t) NESTED TABLE col1 STORE AS col1_tab;
```

Flattening the my_parent_table:

```
select ... from my_parent_table parent, col1_tab child where child.nested_table_id = parent.col1
```

Oracle source data types

The Oracle database for Qlik Replicate supports most Oracle data types. The following table shows the Oracle source data types that are supported when using Qlik Replicate and the default mapping to Qlik Replicate data types.

For information on how to view the data type that is mapped in the target, see the section for the target database you are using.

9 Adding and managing source endpoints


For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).

Data types

Oracle Data Types	Qlik Replicate Data Types
BINARY_FLOAT	REAL4
BINARY_DOUBLE	REAL8
BINARY	BYTES
FLOAT (P)	REAL8
NUMBER (P,S) NUMBER according to the "Expose number as" property in the Qlik Replicate Oracle source database settings.	When scale is < 0: REAL8 When scale is 0 and: Precision = 0: REAL8 Precision < or = 2: INT1 Precision >2 and <or = 4: INT2 Precision >4 and <or = 9: INT4 Precision > 9: NUMERIC If precision > or = scale: NUMERIC In all other cases: REAL8
DATE	DATETIME
INTERVAL_YEAR TO MONTH	STRING (with interval year_to_month indication)
INTERVAL_DAY TO SECOND	STRING (with interval day_to_second indication)
TIMESTAMP	DATETIME
TIMESTAMP WITH TIME ZONE	STRING (with timestamp_with_timezone indication)
TIMESTAMP WITH LOCAL TIME ZONE	STRING (with timestamp_with_local_timezone indication)
CHAR	STRING

9 Adding and managing source endpoints

Oracle Data Types	Qlik Replicate Data Types
VARCHAR2	Length <= 4000 bytes: STRING Length > 4000 bytes: CLOB
NCHAR	Length <= 4000 bytes: WSTRING Length > 4000 bytes: NCLOB
NVARCHAR2	Length <= 4000 bytes: WSTRING Length > 4000 bytes: NCLOB
RAW	Length <= 2000 bytes: BYTES Length > 2000 bytes: BLOB
REAL	REAL8
BLOB To use this data type with Qlik Replicate, you must enable the use of BLOBs for a specific task. During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.	BLOB
CLOB To use this data type with Qlik Replicate, you must enable the use of CLOBs for a specific task. During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.	CLOB

Oracle Data Types	Qlik Replicate Data Types
<p>NCLOB</p> <p>To use this data type with Qlik Replicate, you must enable the use of NCLOBs for a specific task.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p>	NCLOB
<p>LONG</p> <p>The LONG data type is not supported in Batch optimized apply mode.</p> <p>To use this data type with Qlik Replicate, you must enable the use of LOBs for a specific task.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p>	CLOB
<p>LONG RAW</p> <p>The LONG RAW data type is not supported in Batch optimized apply mode.</p> <p>To use this data type with Qlik Replicate, you must enable the use of LOBs for a specific task.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p>	BLOB
<p>XMLTYPE</p> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p> • <i>When replicating XML columns, performance can be improved by not using the Oracle 12 client.</i></p> <p>• <i>Support for the XMLTYPE data type requires the full Oracle Client (as opposed to the Oracle Instant Client).</i></p> </div> <p>When the target column is a CLOB, both full LOB mode and limited LOB mode are supported (depending on the target).</p> <p>For more information, see LOB support in Task Settings/Metadata (page 1166).</p>	CLOB

Unsupported data types

The following data types are not supported:

- BFILE
- ROWID
- REF
- UROWID
- ANYDATA
- SDO_GEOMETRY
- User-defined data types



- *Virtual columns are not supported.*
- *As the ROWID data type is not supported, materialized views based on a ROWID column are also not supported.*

Homogeneous Replication

With the exception of the data types listed in the table below, when replicating from an Oracle source to an Oracle target, all of the source and target data types will be identical.



- *As of Oracle 9.0, the LONG and LONG RAW data types are no longer supported by Oracle.*
- *Primary/Unique Index names are preserved during homogeneous replication.*
- *In homogeneous replication, the source data first passes through the Qlik Replicate data type and is therefore subject to any limitations of that type.*
- *For information on Replicate data types and their limitations (where relevant), see Replicate data types (page 81).*
- *For information on which Replicate data types the source data passes through when replicating from Oracle, see the Oracle to Qlik Replicate data types mapping table described earlier.*

Non-identical data types in homogeneous replication

Oracle Data Types	Qlik Replicate Data Types
VARCHAR2	Length > 4000 bytes: CLOB
NVARCHAR2	Length > 4000 bytes: NCLOB
RAW	Length > 2000 bytes: BLOB

Oracle Data Types	Qlik Replicate Data Types
<p>LONG</p> <p>The LONG data type is not supported in Batch optimized apply mode.</p> <p>To use this data type with Qlik Replicate, you must enable the use of LOBs for a specific task.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p>	<p>CLOB</p>
<p>LONG RAW</p> <p>The LONG RAW data type is not supported in Batch optimized apply mode.</p> <p>To use this data type with Qlik Replicate, you must enable the use of LOBs for a specific task.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p>	<p>BLOB</p>

Working with Oracle on Oracle Cloud

When replicating from Oracle on Oracle Cloud, the following limitations apply:

- Using Oracle LogMiner to access the redo files is not supported.
- Due to insufficient permissions, the following options in the endpoint settings' **Advanced** tab are not supported:
 - Replicate has file-level access to the redo log files
 - Replace path prefix

Working with Amazon RDS for Oracle

Before you can land data from Amazon RDS for Oracle, make sure to fulfill the prerequisites described below.



Oracle databases with Multi-AZ enabled are supported.

Amazon RDS for Oracle considerations and limitations

When replicating from Amazon RDS for Oracle, TDE is supported only with encrypted tablespace and only when using Oracle LogMiner.

Required Amazon RDS for Oracle permissions

This topic describes the permissions required when replicating from Amazon RDS for Oracle.

Grant the following privileges:

- GRANT SELECT ANY TABLE
- GRANT SELECT on ALL_VIEWS to user;
- GRANT SELECT ANY TRANSACTION to user;
- GRANT SELECT on DBA_TABLESPACES to user;
- GRANT SELECT on ALL_TAB_PARTITIONS to user;
- GRANT SELECT on ALL_INDEXES to user;
- GRANT SELECT on ALL_OBJECTS to user;
- GRANT SELECT on ALL_TABLES to user;
- GRANT SELECT on ALL_USERS to user;
- GRANT SELECT on ALL_CATALOG to user;
- GRANT SELECT on ALL_CONSTRAINTS to user;
- GRANT SELECT on ALL_CONS_COLUMNS to user;
- GRANT SELECT on ALL_TAB_COLS to user;
- GRANT SELECT on ALL_IND_COLUMNS to user;
- GRANT SELECT on ALL_LOG_GROUPS to user;
- GRANT LOGMINING TO user;

In addition, execute the following statements:

```
exec rdsadmin.rdsadmin_util.grant_sys_object('V_$ARCHIVED_
LOG', 'user', 'SELECT');
exec rdsadmin.rdsadmin_util.grant_sys_object('V_$LOG', 'user', 'SELECT');
exec rdsadmin.rdsadmin_util.grant_sys_object('V_$LOGFILE', 'user', 'SELECT');
exec rdsadmin.rdsadmin_util.grant_sys_object('V_$DATABASE', 'user', 'SELECT');
exec rdsadmin.rdsadmin_util.grant_sys_object('V_$THREAD', 'user', 'SELECT');
exec rdsadmin.rdsadmin_util.grant_sys_object('V_$PARAMETER', 'user', 'SELECT');
exec rdsadmin.rdsadmin_util.grant_sys_object('V_$NLS_
PARAMETERS', 'user', 'SELECT');
exec rdsadmin.rdsadmin_util.grant_sys_object('V_$TIMEZONE_
NAMES', 'user', 'SELECT');
exec rdsadmin.rdsadmin_util.grant_sys_object('V_
$TRANSACTION', 'user', 'SELECT');
exec rdsadmin.rdsadmin_util.grant_sys_object('V_
$CONTAINERS', 'user', 'SELECT');
exec rdsadmin.rdsadmin_util.grant_sys_object('DBA_REGISTRY', 'user', 'SELECT');
exec rdsadmin.rdsadmin_util.grant_sys_object('OBJ$', 'user', 'SELECT');
exec rdsadmin.rdsadmin_util.grant_sys_object('ALL_ENCRYPTED_
COLUMNS', 'user', 'SELECT');
exec rdsadmin.rdsadmin_util.grant_sys_object('V_$LOGMNR_
LOGS', 'user', 'SELECT');
exec rdsadmin.rdsadmin_util.grant_sys_object('V_$LOGMNR_
CONTENTS', 'user', 'SELECT');
exec rdsadmin.rdsadmin_util.grant_sys_object('DBMS_LOGMNR', 'user', 'EXECUTE');
```

Preparing Replication from Amazon RDS for Oracle

This section details the requirements for working with Oracle on Amazon RDS.

Set up supplemental logging

Qlik Replicate requires database-level supplemental logging to be enabled. To enable database-level supplemental logging, execute the following command:

```
exec rdsadmin.rdsadmin_util.alter_supplemental_logging('ADD');
```

Although not required, examples of additional commands that you can execute to change the supplemental logging attributes include:

```
exec rdsadmin.rdsadmin_util.alter_supplemental_logging('ADD','ALL');  
exec rdsadmin.rdsadmin_util.alter_supplemental_logging('DROP','PRIMARY KEY');
```

Enabling automatic backups

In **Step 5: Management Options** of setting up your Oracle database instance, set the **Enabled Automatic Backups** option to **Yes**.

Setting up archiving

To retain archived redo logs of your Oracle database instance (which will allow Qlik Replicate to retrieve the log information using Oracle LogMiner), execute the following command (example 24 hours):

```
exec rdsadmin.rdsadmin_util.set_configuration('archive_log_retention_hours',24);
```

Make sure that your storage has sufficient space for the archived redo logs during the specified period.

Using Replicate Log Reader to access the redo logs

The following steps are relevant only when using Replicate Log Reader to access the redo logs.

Create directories for online and archived logs

Before you can use Replicate Log Reader, you need to create directories for the online and archived logs.

To create the directories, run the following commands:

```
exec rdsadmin.rdsadmin_master_util.create_archive_log_dir;  
exec rdsadmin.rdsadmin_master_util.create_online_log_dir;
```

To remove the directories (if there are no ongoing replication tasks), run the following commands:

```
exec rdsadmin.rdsadmin_master_util.drop_archive_log_dir;  
exec rdsadmin.rdsadmin_master_util.drop_online_log_dir;
```

Setting the replace path prefix parameter



*Required for Oracle 11.2 and 12.1 only. For all other supported versions, select the Using the path as it appears in the database (page 445) option in the endpoint settings' **Advanced** tab.*

To work with Replicate Log Reader, you must set the **Replace path prefix** option in the **Advanced** tab with the following values:

Replace path prefix: `/rdsdbdata/db/<ORACLE-INSTANCE-NAME>_A/`

With: `/rdsdbdata/log/`

Limitations and considerations

The following Oracle features are not supported when using Replicate Log Reader with Amazon RDS for Oracle:

- ASM
- TDE

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.



- Oracle can also be used as a target database. For information on using Oracle as a target, see *Setting general connection properties (page 971)*.
- You can also use Oracle files as a source or target. For more information, see *Using a file as a source (page 195)*.

To add an Oracle source endpoint to Qlik Replicate

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Oracle database. This is optional.
4. Select **Source** as the database **role**.
5. Select **Oracle** as the database **Type**.



*When this endpoint is used as a duplicated source in a Log Stream Staging setup, select the **Read changes from log stream** check box and then select the relevant Log Stream Staging task from the drop-down list.
For information on setting up and managing Log Stream Staging tasks, see *Using the Log Stream (page 1072)*.*

6. Type the Oracle **Connection String** for the Oracle database you want to work with. You can type the connect string in any Oracle format, for example:

`//host:port/service name`

Where:

- **host:** This is the name or IP address for the computer with the Oracle database that you are using. For example, `johnboy_w7` or `255.255.255.0`.

- **port:** (optional) This is the TNS Listener Port number for the computer with the Oracle database that you are using. If you do not enter a port number the default Oracle TNS Listener port is used.
- **service name:** (optional) This is the service name for the computer with the Oracle database you are using. If you do not enter a service name the default service name is used.

You can also enter an Oracle Net keyword-value pair. For example:

```
"(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP) (HOST=d1sun242) (PORT=5521))
(CONNECT_DATA=(SERVICE_NAME=bjava21)))"
```



When working with a Multitenant environment, the connection string should specify a specific PDB.

7. Type the Oracle authentication information (User Name, Password) for the authorized user for this Oracle database. If you do not know this information, see your Oracle database Administrator (DBA).



This information is case sensitive.



Make sure that the Oracle user entered in the Oracle Authentication section has the correct access privileges. For information on how to provide the required privileges, see [Required permissions \(page 420\)](#).

To ensure that the password provided to Replicate is never used by any other principal, Replicate can be configured to automatically replace the user-entered password with a strong random password. For more information, see [Configuring Replicate to Automatically Replace the User-Entered Password](#).

Working with RAC Instances

This section describes the different procedures that apply when working with RAC instances.

Specifying Separate Connection Strings for Different RAC Instances

If the Oracle endpoint is configured to use [Replicate Log Reader](#) and the node to which Replicate is connected cannot access the logs created by the other cluster nodes, you need to specify a separate connection string for each RAC instance.

When the redo logs are stored in ASM, the connection string syntax is as follows:

```
[<common ASM connection string>,<thread id> <thread ASM connection string>,<thread id>
<thread ASM connection string>...
```



If no <common ASM connection string> is specified, all the RAC instances should be defined in the ASM connection.

When using Replicate Log Reader to access the redo logs, the connection string syntax is as follows:

```
<Oracle connection string>[,<thread id>,<thread BFILE connection string>,<thread id>,<thread BFILE connection string> ...]
```

<Oracle connection string> is mandatory. If specified, the <thread BFILE connection string> will be used instead of the <Oracle connection string>.

Capturing changes after a RAC switchover

If you need to run a switchover procedure on your RAC environment with multiple standby nodes, do the following steps to ensure the continuity of CDC.

1. Run a task using the primary RAC node as a source.
2. Stop the task.
3. Do the switchover procedure. This changes your primary RAC node to a standby role and makes your standby node your primary RAC node.
4. Change the credentials to those of your new primary RAC node.
5. Resume the task. All the events that were triggered when the task was stopped will be replicated automatically and the changes will be captured accordingly.

Setting advanced connection properties

You can configure the Oracle endpoint to access the redo logs using either Oracle LogMiner or Replicate Log Reader, Replicate's high-speed redo log reader.

The following sections describe how to do this:

- *Setting advanced connection properties using Oracle LogMiner (page 442)*
- *Setting advanced connection properties using Replicate Log Reader (page 444)*



For guidelines on choosing which redo logs access method to use, see [Redo log files - access method guidelines \(page 428\)](#).

Setting advanced connection properties using Oracle LogMiner

This section describes which properties are available in the **Advanced** tab when using Oracle LogMiner to access the redo logs. For information on which properties are available in the **Advanced** tab when using Replicate Log Reader to access the redo logs, see *Setting advanced connection properties using Replicate Log Reader (page 444)*.

- **Automatically add supplemental logging:** Select this (the default) to automatically set up supplemental logging for the Oracle database.
For more information on supplemental logging, see *Set up supplemental logging (page 418)*.
- Under the **Access redo logs via** label, choose **Oracle LogMiner**. Changes will be captured using the Oracle LogMiner utility.
Secret Store encryption entries: When some of the source table columns are encrypted you need to specify the Oracle Wallet encryption **Names** and **Values**.

See also: *Finding the wallet entry used for TDE column encryption in a specific table (page 444)* and *Finding the wallet entries used for TDE Encryption (page 450)*.

- **Retry interval:** Use the counter or type the number of seconds that the system waits before resending a query.
- **Archived redo logs destination ID:** The destination of the archived redo logs. The value should be the same as the `DEST_ID` number in the `V$archived_log` table.
When working with multiple log destinations (`DEST_ID`), you should specify an **Archived redo logs location identifier** that represents archived logs that can be accessed by Replicate. If the **Archived redo logs location identifier** is not specified, Replicate will use the ID specified in the **Alternate archived redo logs destination ID** field. If no alternate ID is specified, then Replicate will use the minimal existing `DEST_ID`.
- **Alternate archived redo logs destination ID:** The archive destination in the event of a failure to read from the primary destination.
- **Expose NUMBER as:** Select one of the following:
 - A precision-scale combination
 - FLOAT
 - VARCHAR



- *If precision is 39 or greater, select VARCHAR*
- *By default, the NUMBER data type is converted to precision 38, scale 10*
- *The "Expose NUMBER" definition in the Oracle database is used for the NUMBER data type only (without the explicit precision and scale definition)*

- **Use archived redo logs only:** When this option is selected, Replicate will only access the archived redo logs. If the archived redo logs are stored on ASM only, the Replicate user needs to be granted the ASM privileges described in *Required ASM privileges (page 423)*.
- **Support nested tables:** Select this option if you need to replicate Oracle tables containing columns that are nested tables or defined types. For more information on this feature and its prerequisites, see *Replicating nested tables (page 431)*.
- **Support invisible columns:** Select this option to replicate invisible columns. This option is supported from Oracle 12c and Oracle Instant Client version 12.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

Finding the wallet entry used for TDE column encryption in a specific table

This section describes how to find the correct encryption key used for TDE column encryption in a specific table.

To find the Oracle Wallet entry:

1. On the Oracle database, run the following query to return the `object_id` (e.g. the table ID) according to a given owner and table name:

```
select object_id from all_objects where owner='<table owner>' and object_name='<table name>' and object_type='TABLE';
```
2. Use the retrieved `object_id` in the following query to return the relevant master key:

```
select mkeyid from sys.enc$ where obj#=OBJECT_ID;
```
3. Select the key value from the Oracle Wallet as follows:

```
mkstore -wr1 <full_wallet_name> -viewEntry <entry_name>
```



For more information, see Step 5 in *Finding the wallet entries used for TDE Encryption* (page 450).

4. Copy the master key entry and its value into the **Names** and **Values** fields respectively.

Setting advanced connection properties using Replicate Log Reader

This section describes which properties are available in the **Advanced** tab when using Replicate Log Reader to access the redo logs. For information on which properties are available in the **Advanced** tab when using LogMiner to access the redo logs, see *Setting advanced connection properties using Oracle LogMiner* (page 442).

- **Automatically add supplemental logging:** Select this to automatically set up supplemental logging for the Oracle database. This option is also available when LogMiner is selected as the redo logs access method.
For more information on supplemental logging, see *Set up supplemental logging* (page 418).
- Under the **Access redo logs via** label, choose **Replicate Log Reader** (the default). Replicate will access the redo logs as a binary file.
- **Secret Store encryption entries:** When the source tables are encrypted or contain encrypted columns, you need to specify the Oracle Wallet encryption keys and their values.
For information on locating the required keys, see *Finding the wallet entries used for TDE Encryption* (page 450).

- **ASM Parameters (if redo logs are stored in ASM)** - If the Oracle redo logs you are using are stored using Automated Storage Management (ASM), enter the required access information in the designated fields.



- *These options are only available when **Access redo logs via Log Reader** is selected.*
- *To access the redo logs in ASM, you also need to grant the additional privileges described in [Required ASM privileges \(page 423\)](#)*

- **ASM Connection String:** The connection string to the ASM instance if your Oracle database is using ASM.
- **ASM user name:** The user name for the ASM user.
- **ASM password:** The password for the ASM user.
- **Number of concurrent ASM Read threads:** The number of threads to use when reading the redo logs from ASM. Increasing this number may improve performance when there is a large volume of changes.
- To access a redo log as a binary file (i.e. not using LogMiner), select one of the following options:
 - **Use path as it appears in the database:** Select this to access the redo logs using the path as it appears in the database. Continue from [Using the Path as it Appears in the Database](#).
 - OR-
 - **Replace path prefix:** You can determine whether to read the redo logs from a different root location while leaving the relative path unchanged. Continue from [Replacing the Path with a Prefix](#).

Using the path as it appears in the database

- **Replicate has file-level access to the redo log files:** Select this to access and read the redo logs directly from the file system of the local computer where Qlik Replicate is installed.
- **Copy redo logs to a temporary folder:** Select this to copy the redo logs to a temporary folder and then specify the path of the redo logs on the Oracle machine.



*When configuring multiple tasks that use the same temporary folder (configured in the Oracle source endpoint), do not select the **Delete processed archived redo log files** option. This is because Replicate uses the original archived log names.*



When working in a RAC environment, it is strongly recommended to set up a shared folder that is accessible by all the RAC instances. If this is not possible, you need to define a temporary folder with the same name on each of the RAC instances. In addition, you need to define separate Oracle and ASM connection strings for each RAC instance. For more information on defining RAC connection strings, see [Setting general connection properties \(page 440\)](#).

- **Replicate has file-level access to temporary folder:** Select this to access the archived redo logs directly from the file system of the local computer where Qlik Replicate is installed.

- **Access archived redo logs in folder:** To enable Qlik Replicate to access the temporary folder (when it has file level access), you need to specify the path to the shared temporary folder on the Oracle machine, e.g. \\my.oracle.box\tempshare.



When a stopped task is resumed, Replicate will try to re-copy the currently processed Redo logs. If there are no Redo logs in the specified directory, the task will wait for them to be copied there.

- **Look for missing archived redo logs in folder:** Type the full path to a location from where you want Qlik Replicate to read the archived redo logs if they are not found in the default location. The folder can be located anywhere in the network where Qlik Replicate is located, but be sure that the location is accessible to the Qlik Replicate user.



*When this option is configured, the name of each redo log in the folder must be identical to the name returned by `v$archived_log` view for the current sequence, thread, and destination ID.
As the `v$archived_log` view returns redo log names only when the log status is ACTIVE (meaning the log exists), Replicate will be unable to determine which redo log to access when the status is not ACTIVE.*

- **Replicate has file-level access to the specified folder:** Select this to access and read the archived redo logs directly from the file system of the local computer where Qlik Replicate is installed.
- **Delete processed archived redo log files:** Select this to delete the copied archived redo log files after they have been read.



*Archived redo logs will only be deleted if the **Copy redo logs to temporary folder** option is enabled as well.*

This option requires the following additional permissions for the Replicate user:

- GRANT SELECT ON DBA_FILE_GROUPS
Example:
`GRANT SELECT ON DBA_FILE_GROUPS to nonpriv_user;`
- GRANT EXECUTE on SYS.DBMS_FILE_GROUP
Example:
`GRANT EXECUTE ON SYS.DBMS_FILE_GROUP to nonpriv_user;`
- EXECUTE DBMS_FILE_GROUP.GRANT_SYSTEM_PRIVILEGE with the system privilege 'MANAGE_FILE_GROUP' for the Replicate user.
Example:
`execute DBMS_FILE_GROUP.GRANT_SYSTEM_PRIVILEGE (DBMS_FILE_GROUP.MANAGE_FILE_GROUP, 'nonpriv_user', FALSE)`



Verify that another file group is not using the configured temp directory under a different Oracle user.

- **Retry interval:** Use the counter or type the number of seconds that the system waits before resending a query.
- **Archived redo logs destination ID:** The destination of the archived redo logs. The value should be the same as the `DEST_ID` number in the `V$archived_Log` table.
When working with multiple log destinations (`DEST_ID`), you should specify an **Archived redo logs location identifier** that represents archived logs that can be accessed by Replicate. If the **Archived redo logs location identifier** is not specified, Replicate will use the ID specified in the **Alternate archived redo logs destination ID** field. If no alternate ID is specified, then Replicate will use the minimal existing `DEST_ID`.
- **Alternate archived redo logs destination ID:** The archive destination in the event of a failure to read from the primary destination.
- **Expose NUMBER as:** Select one of the following:
 - A precision-scale combination
 - FLOAT
 - VARCHAR



- *If precision is 39 or greater, select VARCHAR*
- *By default, the NUMBER data type is converted to precision 38, scale 10*
- *The "Expose NUMBER" definition in the Oracle database is used for the NUMBER data type only (without the explicit precision and scale definition)*

- **Use archived redo logs only:** When this option is selected, Replicate will only access the archived redo logs. If the archived redo logs are stored on ASM only, the Replicate user needs to be granted the ASM privileges described in *Required ASM privileges (page 423)*.
- **Support nested tables:** Select this option if you need to replicate Oracle tables containing columns that are nested tables or defined types. For more information on this feature and its prerequisites, see *Replicating nested tables (page 431)*.
- **Support invisible columns:** Select this option to replicate invisible columns. This option is supported from Oracle 12c and Oracle Instant Client version 12.

Replacing the path prefix

- **Replace path prefix:** You can determine whether to read the redo logs from a different root location while leaving the relative path unchanged.
Type the first part of the path to the current location of the redo logs. For example, `C:\OldFolder`. You can include one folder or directory level or multiple folders or directories in this field.
With: Type the name of the folder or prefix to replace the existing prefix that you added in the field above. For example, `C:\NewFolder`.



The following examples illustrate how to change the prefix:

If the redo logs are located in `C:\OldFolder\archive\logs` and you specify `C:\OldFolder` in the **Replace path prefix** field, and `C:\NewFolder` in the **With** field, the redo logs will be read from:

`C:\NewFolder\archive\logs`

If the redo logs are located in `C:\temp\oracle\logs\archive\RedoLogs` and you specify

`C:\temp\oracle\logs` in the **Replace path prefix** field, and `C:\companyName` in the **With** field, then the redo logs will be read from:

`C:\companyName\archive\RedoLogs`

In this case, the new folder or directory called `companyName` replaces all of the first three level folders that you included in the **Replace path prefix** field.

- **Apply prefix replacement to online and archived redo logs:** Select this to apply the prefix replacement to the online and archived redo logs.
 - **Replicate has file-level access to the new location:** Select this to access and read the online and archived redo log files directly from the file system of the local computer where Qlik Replicate is installed.
- **Apply prefix replacement to archived redo logs only:** Select this to apply the prefix replacement to the archived redo logs only (and not to the online redo logs).
 - **Replicate has file-level access to the original online location:** Select this to access and read the original online redo log files directly from the file system of the local computer where Qlik Replicate is installed.
 - **Replicate has file-level access to the new archive location:** Select this to access and read the archived redo log files directly from the file system of the local computer where Qlik Replicate is installed.
- **Delete processed archived redo log files:** Select this to delete the copied archived redo log files after they have been read.



Archived redo logs will only be deleted if the **Copy redo logs to temporary folder** option is enabled as well.

This option requires the following additional permissions for the Replicate user:

- GRANT SELECT ON DBA_FILE_GROUPS
Example:
`GRANT SELECT ON DBA_FILE_GROUPS to nonpriv_user;`
- GRANT EXECUTE on SYS.DBMS_FILE_GROUP
Example:
`GRANT EXECUTE ON SYS.DBMS_FILE_GROUP to nonpriv_user;`
- EXECUTE DBMS_FILE_GROUP.GRANT_SYSTEM_PRIVILEGE with the system privilege 'MANAGE_FILE_GROUP' for the Replicate user.
Example:
`execute DBMS_FILE_GROUP.GRANT_SYSTEM_PRIVILEGE (DBMS_FILE_GROUP.MANAGE_FILE_GROUP, 'nonpriv_user', FALSE)`



Verify that another file group is not using the configured temp directory under a different Oracle user.

- **Retry interval:** Use the counter or type the number of seconds that the system waits before resending a query.
- **Archived redo logs destination ID:** The destination of the archived redo logs. The value should be the same as the `DEST_ID` number in the `V$archived_log` table. When working with multiple log destinations (`DEST_ID`), you should specify an **Archived redo logs location identifier** that represents archived logs that can be accessed by Replicate. If the **Archived redo logs location identifier** is not specified, Replicate will use the ID specified in the **Alternate archived redo logs destination ID** field. If no alternate ID is specified, then Replicate will use the minimal existing `DEST_ID`.
- **Alternate archived redo logs destination ID:** The archive destination in the event of a failure to read from the primary destination.
- **Expose NUMBER as:** Select one of the following:
 - A precision-scale combination
 - FLOAT
 - VARCHAR



- *If precision is 39 or greater, select VARCHAR*
- *By default, the NUMBER data type is converted to precision 38, scale 10*
- *The "Expose NUMBER" definition in the Oracle database is used for the NUMBER data type only (without the explicit precision and scale definition)*

- **Use archived redo logs only:** When this option is selected, Replicate will only access the archived redo logs. If the archived redo logs are stored on ASM only, the Replicate user needs to be granted the ASM privileges described in *Required ASM privileges (page 423)*.
- **Support nested tables:** Select this option if you need to replicate Oracle tables containing columns that are nested tables or defined types. For more information on this feature and its prerequisites, see *Replicating nested tables (page 431)*.
- **Support invisible columns:** Select this option to replicate invisible columns. This option is supported from Oracle 12c and Oracle Instant Client version 12.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link. The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.

4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

Finding the wallet entries used for TDE Encryption

In order to specify the correct encryption key(s) used for TDE tablespace encryption or TDE column encryption, you first need to find the relevant entry (or entries in the case of multiple keys) in the Oracle Wallet containing the encryption key(s). After you find the relevant entry or entries, copy the entry and its value (or entries and values if more than one) into the **Names** and **Values** fields respectively.



*To enter multiple values, first copy each entry into a text editor such as Notepad making sure to separate the values with a comma. Then, copy the string containing the values and commas from the text editor and paste it into the **Values** field. There is no need to do this for entries. You can paste the entries directly into the **Entries** field, remembering to separate each entry with a comma.*


To find the Oracle Wallet entries:


1. If the ENCRYPTION_WALLET_LOCATION parameter is defined in the sqlnet.ora file, use the wallet from the directory defined by this parameter.
2. If the WALLET_LOCATION parameter is defined in the sqlnet.ora file, use the wallet from the directory defined by this parameter.
3. In other cases, use the wallet in the default database location.



*The name of the wallet should be **ewallet.p12***

4. Use the "list" option in the Oracle mkstore utility to determine the ORACLE.SECURITY.DB/TS.ENCRYPTION.<SUFFIX> entry name(s), as follows:
`mkstore -wrl <full wallet name> -list`
5. If you know which entry/entries is/are used to encrypt the Redo logs, select the entry name(s) and use the "viewEntry" option in the Oracle mkstore utility to determine the entry value, as follows:
`mkstore -wrl <full wallet name> -viewEntry <entry name>`

 If you do not know which entry is used to encrypt the Redo logs, you can select multiple DB or TS entries and determine their values as described above (and then copy and paste the entry names and values into the **Names** and **Values** fields as described in the Finding the wallet entries used for TDE Encryption (page 450)). If the specified entries are not correct, the task will fail and the error message will contain the correct entry name.

 If the DBA changes the entry while the task is running, the task will fail and the error message will contain the new entry name. Add the new entry (name and value) to the already specified entries and then resume the task.

9.25 Using PostgreSQL as a source

This section describes how to set up and use a PostgreSQL database as a source in a replication task.



The PostgreSQL source endpoint also supports replication from Microsoft Azure Database for PostgreSQL - Flexible Server.

In this section:

- *Prerequisites (page 453)*
- *Required permissions (page 454)*
- *Using an account without the "superuser" role (page 454)*
- *Limitations and considerations (page 456)*
- *Supported data types (page 457)*
- *Homogeneous replication (page 461)*
- *Setting general connection properties (page 463)*
- *Setting advanced connection properties (page 464)*
- *Capturing changes from Microsoft Azure for PostgreSQL (page 466)*
- *Removing Replicate artifacts from the database (page 466)*

Prerequisites

The following section describes the prerequisites for working with the Qlik Replicate PostgreSQL source endpoint.

Client side

Replicate for Windows

Install PostgreSQL ODBC Driver `psqlodbc_13_xx_xxxx-x64` on the Replicate machine.



Make sure to add the `psql.exe` path (for example, `"C:\Program Files\PostgreSQL\13xx\bin"`) to the system `PATH`.

Replicate for Linux

Download PostgreSQL 13 ODBC driver, and client programs and libraries.

- **Example PostgreSQL ODBC file name:** `postgresql13-odbc-13.02.0000-1PGDG.rhel8.x86_64.rpm`
- **PostgreSQL client programs and libraries file name:** `postgresql13-13.8-1PGDG.rhel<version>.x86_64.rpm`

Then, open a shell prompt and do the following:

1. Stop the Replicate service:
`./areplicate stop`
2. Optionally, confirm that the service has stopped:
`./areplicate status`
3. Install PostgreSQL 13 client programs and libraries.
4. Install the ODBC driver for PostgreSQL 13.
5. Change the working directory to `<Replicate-Install-Dir>/bin`.
6. Copy the driver location to the `site_arep_login.sh` file:
`echo "export LD_LIBRARY_PATH=\$LD_LIBRARY_PATH:/usr/lib:/usr/lib64:/usr/pgsql-13/lib" >> site_arep_login.sh`
This will add the driver to "LD_LIBRARY_PATH" and update the driver location in the **site_arep_login.sh** file.
7. Optionally, confirm that the driver location was copied:
`cat site_arep_login.sh`
8. Makes sure that the `/etc/odbcinst.ini` file contains an entry for PostgreSQL, as in the following example:

```
[PostgreSQL Unicode(x64)]
Description = PostgreSQL ODBC driver
Driver = /usr/pgsql-13/lib/psqlodbc.so
Setup = /usr/pgsql-13/lib/psqlodbcw.so
UsageCount = 1
```
9. Start the Replicate service:
`./areplicate start`
10. Optionally, confirm that the service has started:
`./areplicate status`

Server side

- The IP address of the Replicate machine must be added to the `pg_hba.conf` configuration file with the "replication" keyword in the database field.

Example:

```
host replication all 176.123.1.212/32 trust
```

- Make sure that the **test_decoding** output plugin (found in the **postgresql12-contrib** package) is installed.
- The user specified in the connection settings must be granted read-write permissions to the source database to allow the creation of "slots" on the database. The slots are required in order to read the changes.
- The following parameters and values must be set in the `postgresql.conf` configuration file.
`wal_level = logical`
`max_replication_slots = number of replication slots`
Where *number of replication slots* must be greater than one and should be set according to the number of tasks that you want to run. For example, to run five tasks you need to set `max_replication_slots = 5`. Slots open automatically as soon as a task starts and remain open, even when task is no longer running. Note that open slots need to be manually deleted.
`max_wal_senders = number of concurrent tasks`
Where *number of concurrent tasks* must be greater than one and should be set according to the number of tasks that you are running concurrently. For example, to run three tasks concurrently you need to set `max_wal_senders = 3`.
- The `wal_sender_timeout` parameter terminates connections that are inactive longer than the specified number of milliseconds. The default timeout is 60 seconds. To disable the timeout mechanism (optional), set this parameter to zero.



By default, the value of the `wal_sender_timeout` parameter is interpreted by the server as milliseconds. To explicitly specify seconds, append an "s" to the value as in the following example:

```
wal_sender_timeout=60s
```

For more information on the configuration parameters, see the Replication section under Server Configuration in the [PostgreSQL online help](#).

Required permissions

The user specified in the endpoint connection settings must be granted the following permissions on the PostgreSQL database:

- **For Full Load tasks:** Standard SELECT
- **For CDC tasks:** Superuser
See also *Using an account without the "superuser" role* (page 454).

Using an account without the "superuser" role

If you are not using an account with the **superuser** role, you need to create several objects to capture Data Definition Language (DDL) events. Create these objects in the selected account and then create a trigger in the master user account.

To do this:

1. Choose a schema where you want the objects to be created. The default schema is public. Ensure that the schema exists and is accessible by the non-privileged account.

2. Create the table **attrep_ddl_audit** by running the following command:

```
create table <objects_schema>.attrep_ddl_audit
(
  c_key      bigserial primary key,
  c_time     timestamp,      -- Informational
  c_user     varchar(64),    -- Informational: current_user
  c_txn      varchar(16),    -- Informational: current transaction
  c_tag      varchar(24),    -- Either 'CREATE TABLE' or 'ALTER TABLE' or
  'DROP TABLE'
  c_oid      integer,       -- For future use - TG_OBJECTID
  c_name     varchar(64),    -- For future use - TG_OBJECTNAME
  c_schema   varchar(64),    -- For future use - TG_SCHEMA_NAME. For now,
  holds the current_schema
  c_ddlqry   text           -- The DDL query associated with the current
  DDL event
);
```

3. Create the function **attrep_intercept_ddl** by running the following command:

```
CREATE OR REPLACE FUNCTION <objects_schema>.attrep_intercept_ddl()
  RETURNS event_trigger
LANGUAGE plpgsql
  AS $$
  declare _qry text;
BEGIN
  if (tg_tag='CREATE TABLE' or tg_tag='ALTER TABLE' or tg_tag='DROP
  TABLE') then
    SELECT current_query() into _qry;
    insert into <objects_schema>.attrep_ddl_audit
    values
      (
        default,current_timestamp,current_user,cast(TXID_CURRENT() as
        varchar(16)),tg_tag,0,'',current_schema,_qry
      );
    delete from <objects_schema>.attrep_ddl_audit;
  end if;
END;
$$;
```

4. If you are logged in with a non-privileged account, log out of the non-privileged account and log in with an account that has the **superuser** role assigned to it.



If the **attrep_intercept_ddl** stored procedure is not being created in the default schema, you need to specify the schema name in the **Create DDL artifacts in schema** field in the endpoint settings' **Advanced** tab.

For more information on the replication configuration parameters, see the PostgreSQL Help.

5. Create the event trigger **attrep_intercept_ddl** by running the following command:

```
CREATE EVENT TRIGGER attrep_intercept_ddl ON ddl_command_end
EXECUTE PROCEDURE <objects_schema>.attrep_intercept_ddl();
```

6. Grant the following permissions to the non-privileged account:

- GRANT INSERT ON attrep_ddl_audit to <non-privileged-user>;
- GRANT DELETE ON attrep_ddl_audit to <non-privileged-user>;
- GRANT USAGE ON attrep_ddl_audit_c_key_seq TO <non-privileged-user>;
- ALTER ROLE <non-privileged-user> WITH REPLICATION;

7. Grant SELECT permission to the user:

```
GRANT SELECT ON ALL TABLES IN SCHEMA <SCHEMA_NAME> TO <non-privileged-user>;
```

8. Grant usage on the schema to the user:

```
GRANT USAGE ON SCHEMA <SCHEMA_NAME> TO <non-privileged-user>;
```

Limitations and considerations

You should be aware of the following limitations and considerations when using a PostgreSQL source:

- The database name cannot include a semi-colon (;).
- Capturing changes while the database is in Hot Standby mode is not supported.
- To capture changes from a secondary database, read/write privileges must be configured on the database. Due to the write permission requirement, the PostgreSQL read replica (in a High Availability environment) is not supported.
For information about the read-write permission, see *Prerequisites (page 453)*
- Replication of multiple tables with the same name but a different case (e.g. table1, TABLE1 and Tab1e1) may cause unpredictable behavior and is therefore not supported.
- Change processing of TRUNCATE operations is not supported.
- Deferred Constraints are not supported.
- The default configuration of Replica identity is supported only.
For more information about Replica identity, see <https://www.postgresql.org/docs/10/logical-replication-publication.html>
- Unique Indexes with CASE expressions are not supported.
- Both the source table and the corresponding target table must have an identical Primary Key. In the event that one of the tables does not have a Primary Key, the result of DELETE and UPDATE record

operations will be unpredictable.

- The “Start Process Changes from Timestamp” run option is not supported.
- Replication of the Before Image is not supported.
- Change processing of [CREATE | ALTER | DROP] table DDLs are supported unless they are held in an inner function/procedure body block or in other nested constructs.

For example, the following change will not be captured:

```
CREATE OR REPLACE FUNCTION attu.create_distributors1() RETURNS void
LANGUAGE plpgsql
AS $$
BEGIN
    create table attu.distributors1(did serial PRIMARY KEY,name varchar(40) NOT NULL);
END;
$$;
```

- **Partitioned tables**
 - CDC is not supported.
- When using the [Parallel Load](#) feature, table segmentation according to partitions or sub-partitions is not supported.
- When using [filters](#) that do not contain a Primary Key, DELETE operations will not be captured.
- A DDL operation that occurs during Full Load, but that was not preceded by a DML operation will not be captured.
- PostGIS data types are not supported.
- Forced failover is a feature of Azure Database for PostgreSQL – Flexible Server. If forced failover occurs when replicating data from Azure Database for PostgreSQL – Flexible Server, the task will stop and will need to be reloaded.
- The SQL MERGE command introduced in PostgreSQL 15.x has not been certified.

Unsupported data types



The PostGIS data type is not supported.

Supported data types

The following table shows the PostgreSQL source data types that are supported when using Qlik Replicate and the default mapping to the Qlik Replicate data types.


For information about data type mapping in homogeneous replication, see *Homogeneous replication* (page 461).



For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).

Data types



PostgreSQL Data Types	Qlik Replicate Data Types
INTEGER	INT4

9 Adding and managing source endpoints

PostgreSQL Data Types	Qlik Replicate Data Types
SMALLINT	INT2
BIGINT	INT8
NUMERIC	NUMERIC(28,6)
NUMERIC(p,s)	<p>If precision is ≤ 38, then:</p> <p>NUMERIC(p,s)</p> <p>If precision is ≥ 39, then:</p> <p>STRING</p>
DECIMAL	NUMERIC(28,6)
DECIMAL(p,s)	<p>If precision is ≤ 38, then:</p> <p>NUMERIC(p,s)</p> <p>If precision is ≥ 39, then:</p> <p>STRING</p>
REAL	REAL4
DOUBLE	REAL8
SMALLSERIAL	INT2
SERIAL	INT4
BIGSERIAL	INT8
MONEY	<p>NUMERIC(38,4)</p> <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;"> <p> <i>The MONEY data type is mapped to FLOAT in Microsoft SQL Server.</i></p> </div>
CHAR	WSTRING (1)
CHAR(n)	WSTRING (n)

PostgreSQL Data Types	Qlik Replicate Data Types
VARCHAR(n) <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p> <i>VARCHAR without a length (n) is not recognized as a valid data type by target endpoints. Consequently, if a source column data type is set to VARCHAR without an explicit length, Replicate will set a default length of 8000 bytes.</i></p> <p><i>You can change the default by setting the following internal parameter to the required length:</i></p> <p style="margin-left: 20px;"><code>unboundedVarcharMaxSize</code></p> <p><i>See also Internal Parameters.</i></p> </div>	WSTRING (n)
TEXT	NCLOB
BYTEA	BLOB
TIMESTAMP	DATETIME
TIMESTAMP (z)	DATETIME
DATE	DATE
<div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p> <i>Replicate only supports ISO formatted textual DATE formats (the default). If other formats are used, an error will be generated. You can change the date format in the postgresql.conf file or using the PGDATESTYLE environment variable. You can also change the date format at database level.</i></p> </div>	
TIME	TIME
TIME (z)	TIME
INTERVAL	STRING (128) - 1 YEAR, 2 MONTHS, 3 DAYS, 4 HOURS, 5 MINUTES, 6 SECONDS
BOOLEAN	STRING (5)
ENUM	STRING (64)
CIDR	STRING (50)
INET	STRING (50)
MACADDR	STRING (18)
BIT (n)	STRING (n)

9 Adding and managing source endpoints

PostgreSQL Data Types	Qlik Replicate Data Types
BIT VARYING (n)	STRING (n)
UUID	STRING
TSVECTOR	CLOB
TSQUERY	CLOB
XML	CLOB
POINT	STRING (255) "(x,y)"
LINE	STRING (255) "(x,y,z)"
LSEG	STRING (255) "((x1,y1), (x2,y2))"
BOX	STRING (255) "((x1,y1), (x2,y2))"
PATH	CLOB "((x1,y1), (xn,yn))"
POLYGON	CLOB "((x1,y1), (xn,yn))"
CIRCLE	STRING (255) "(x,y,r)"
JSON	NCLOB
<p> Reading and writing JSON data is performed using wide string (where each character occupies 2 bytes). Therefore, when replicating JSON columns to a PostgreSQL target, set the limited lob size to twice the maximum JSON length. So, for example, if the JSON data is 4 KB, set the limited lob size value to 8 KB.</p>	
JSONB	NCLOB
<p> Reading and writing JSONB data is performed using wide string (where each character occupies 2 bytes). Therefore, when replicating JSONB columns to a PostgreSQL target, set the limited lob size to twice the maximum JSONB length. So, for example, if the JSONB data is 4 KB, set the limited lob size value to 8 KB.</p>	
ARRAY	NCLOB
COMPOSITE	NCLOB
INT4RANGE	STRING (255)

PostgreSQL Data Types	Qlik Replicate Data Types
INT8RANGE	STRING (255)
NUMRANGE	STRING (255)
TSRANGE	STRING (255)
INT4MULTIRANGE	STRING (255)
INT8MULTIRANGE	STRING (255)
NUMMULTIRANGE	STRING (255)
TSMULTIRANGE	STRING (255)
CHARACTER VARYING	If length is specified: WSTRING (LENGTH) If no length is specified: WSTRING (8000)
TINTERVAL	WSTRING(255)



The PostGIS data type is not supported.

Homogeneous replication

The following section describes how Replicate handles replication from PostgreSQL to the following targets: PostgreSQL, Microsoft Azure Database for PostgreSQL, Google Cloud SQL for PostgreSQL, and Google Cloud AlloyDB for PostgreSQL.

In homogeneous replication, source column and table collations will be replicated to the target as described in *Column and table collation* (page 462).



In homogeneous replication, the source data first passes through the Qlik Replicate data type and is therefore subject to any limitations of that type.

*For information on Replicate data types and their limitations (where relevant), see *Replicate data types* (page 81).*

*For information on which Replicate data types the source data passes through when replicating from PostgreSQL, see the *PostgreSQL to Qlik Replicate data types mapping table* described earlier.*

Limitations and considerations


Homogeneous replication is not supported with the following Replicate configurations or operations:

- Apply Changes only tasks
- Stopping and resuming a task
- Starting a task from a timestamp
- When the **If target table already exists** in the task settings' **Full Load Settings** tab is set to **Do nothing**

Data type considerations and exceptions

When replicating from PostgreSQL to a PostgreSQL-based target, source and target data types are identical for all supported PostgreSQL versions, with the following exceptions:

Data type exceptions in homogeneous replication

PostgreSQL Source	PostgreSQL Target
ENUM	STRING
COMPOSITE	STRING
NUMERIC	NUMERIC (28,6)
NUMERIC (p,s)	NUMERIC
A numeric array with precision and scale.	A numeric array without precision and scale.
TINTERVAL	 <p><i>As PostgreSQL have discontinued support for the TINTERVAL data type, tables with this data type will not be created on the target if the target database version does not support it.</i></p>
INT4MULTIRANGE	STRING (255)
INT8MULTIRANGE	STRING (255)
NUMMULTIRANGE	STRING (255)
TSMULTIRANGE	STRING (255)

See also the note about JSON data type replication in *Supported data types (page 457)*.

Column and table collation

When replicating from PostgreSQL to a PostgreSQL-based target, column and table collations will be replicated to the target.



To support collation replication, the DBA must ensure that the collations defined for the source PostgreSQL database are the same as those defined for the PostgreSQL-based target database.

Non-nullable columns and primary/unique index names

Non-nullable columns and Primary/Unique Index names are preserved during homogeneous replication.

Replication of Partitioned Tables

In order to replicate partitioned tables from PostgreSQL to a PostgreSQL-based target, you need to do the following:

1. Manually create the parent and child tables in the target database.
2. Define a separate task to replicate to those tables.
3. Set the **Truncate before loading** option in the *Full Load Settings* (page 1171).

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a PostgreSQL endpoint source database to Qlik Replicate:

1. In the Qlik Replicate console, click **Manage Endpoint Connections** to open the **Manage Endpoint Connections** dialog box. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints* (page 114).
2. In the **Name** field, type a name for your PostgreSQL database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the PostgreSQL database. This is optional.
4. Select **Source** as the database **role**.
5. Select **PostgreSQL** as the database **Type**.



*When this endpoint is used as a duplicated source in a Log Stream Staging setup, select the **Read changes from log stream** check box and then select the relevant Log Stream Staging task from the drop-down list.*

*For information on setting up and managing Log Stream Staging tasks, see *Using the Log Stream* (page 1072).*

6. Type the **Server** name. This is the name or IP address of the computer with the PostgreSQL database that you want to access.
7. Optionally, change the default port (5432).
8. Enter the PostgreSQL database authentication information (**User name**, **Password**) of an authorized PostgreSQL user.



Consider the following:

- This information is case sensitive.
- To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**. If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box. To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

9. Type the **Database** name or browse for a source database.

SSL authentication options

In the **Security** section, you can configure Replicate to connect to the PostgreSQL database using SSL.

- **SSL Mode:** Select one of the following:
 - **Disable** - Connect with a surname and password only.
 - **Allow** - Establish an encrypted connection if requested by the server.
 - **Prefer** - Establishes an encrypted connection if the server supports encrypted connections, falling back to an unencrypted connection if an encrypted connection cannot be established.
 - **Require** - Establishes an encrypted connection if the server supports encrypted connections. The connection attempt fails if an encrypted connection cannot be established.
 - **Verify CA** - Similar to **Required**, but also verifies the server Certificate Authority (CA) certificate against the configured CA certificates. The connection attempt fails if no valid matching CA certificates are found.
 - **Verify Full** - Similar to **Verify CA**, but also performs host name identity verification by checking the host name the client (i.e. Replicate) uses for connecting to the server against the identity in the certificate that the server sends to the client. The client checks whether the host name that it uses for connecting matches the Common Name value in the server certificate. The connection fails if there is a mismatch.
- **Client certificate path** - The path to the client certificate requested by the server.
- **Secret key path** - The path to the client private key file in PEM format.
- **CA path** - The path to the Certificate Authority (CA) that issued the client certificate file in PEM format.
- **CRL path** - The path to the CRL certificate. This file contains certificates revoked by certificate authorities. If the server certificate appears in this list, the connection will fail.
- **SSL compression** - Select this option to compress the data before it is encrypted.

Setting advanced connection properties

In the **Advanced** tab, you can set the following properties:

- **Capture DDLs:** When this option is selected, the following actions occur:
 - Operational artifacts are created (by Replicate) in the database when the task starts. In order to capture DDL events, Qlik Replicate creates various artifacts in the PostgreSQL database when the task starts. You can later remove these artifacts as described in *Removing Replicate artifacts from the database (page 466)*.
 - Streamed DDL events are captured.
- **Create DDL artifacts in schema:** The schema in which the operational DDL database artifacts will be created. The default value is "Public".
- **WAL heartbeat** - An Apply Changes task that is running but not capturing changes (due to source table inactivity) will continue to occupy the LSN position in its replication slot, thereby preventing truncation of the WAL. Since the WAL is a server-wide resource used by all PostgreSQL processes, it may grow extremely large if no changes are captured for an extended period. To prevent this from happening, enable the "WAL heartbeat" option. When this option is enabled, the PostgreSQL source endpoint mimics task activity by periodically committing pseudo transactions (i.e. "Heartbeats") to the heartbeat table, thereby advancing the task slot's LSN position.
 - **Create WAL heartbeat table in schema:** The schema in which the WAL heartbeat table (`attrep_wal_heartbeat`) will be created. The default value is "public".
 - **Heartbeat frequency (minutes):** The frequency with which to commit transactions to the heartbeat table.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

Capturing changes from Microsoft Azure for PostgreSQL

To be able to capture changes from Microsoft Azure for PostgreSQL, you need to set the "replication" property to "logical", as described in the steps below.

1. Open the Azure CLI and run the following command:



You can download the CLI or connect via a shell command on Azure portal.

Syntax:

```
az postgres server configuration set --resource-group resource-group --server-name server-name --name azure.replication_support --value logical
```

Example:

```
az postgres server configuration set --resource-group azuredb --server-name qlik-azure-pg --name azure.replication_support --value logical
```

2. Restart the database (using the command line or via the portal).

Removing Replicate artifacts from the database

In order to capture DDLs, Qlik Replicate creates various artifacts in the PostgreSQL database when the task starts. When the task completes, you may wish to remove these artifacts.

To remove the artifacts, issue the following statements (in the order they appear below), where `public` is the default schema in which the artifacts were created:

- `drop event trigger attrep_intercept_ddl`



The event trigger does not belong to a specific schema.

- `drop function public.attrep_intercept_ddl()`
- `drop table public.attrep_ddl_audit`
- `drop schema public`
- `drop table schema.attrep_wal_heartbeat`

where `schema` is the **Schema for heartbeat artifacts** specified in the **Advanced** tab of the endpoint settings.



Dropping a schema should be done with extreme caution, if at all. Never drop an operational schema, especially not `public`.

9.26 Using Salesforce (Streaming CDC) as a source

This section describes how to set up and use the Salesforce (Streaming CDC) source endpoint in a replication task. The Salesforce (Streaming CDC) source endpoint is similar to the Salesforce (Incremental Load) source endpoint, although the method each uses to capture changes is different.

Salesforce (Streaming CDC) uses the Salesforce Streaming API which is [subject to several limitations](#), although it does provide better performance. Salesforce (Incremental Load), however, is based on change polling which, although slower than Salesforce (Streaming CDC), has the advantage of supporting almost all tables.

Basically, if you need to capture changes from tables that are not supported with Salesforce (Streaming CDC) then choose Salesforce (Incremental Load). Otherwise, choose Salesforce (Streaming CDC) for its better performance during change capture.

In this section:

- *Prerequisites (page 468)*
- *Calculating the Salesforce quota (page 468)*
- *Limitations and considerations (page 470)*
- *Supported data types (page 471)*
- *Setting general connection properties (page 472)*
- *Setting advanced connection properties (page 473)*

Prerequisites

The following section describes the prerequisites for replicating from a Salesforce database.

- The Salesforce endpoint supports replication of standard objects enabled for replication and enterprise custom objects.
To enable replication of custom objects, the following permissions should be set on the object's detail page:

- Allow Sharing
- Allow Bulk API Access - Only required when [using the Bulk API for Full Load](#)
- Allow streaming API Access - Only required for capturing changes

- **For Apply Changes tasks:**

The Replicate user must be granted the following permissions on the Salesforce database:

- View All Data
- View all Users

As Replicate interfaces with the Salesforce API, each object that you wish to capture must be added to the selected entities in the Change Data Capture API page.

- To work with the Salesforce endpoint, you must install the following API versions:
 - SOAP API - 53.0
 - Bulk API - 53.0
 - Streaming API - 43.0

Calculating the Salesforce quota

The Salesforce endpoint leverages Salesforce APIs to provide Full Load and Change Processing replication functionality. When using the Salesforce endpoint, it's important to be aware of the API quota limits in order to plan your replication tasks accordingly.

Replicate utilizes the following Salesforce APIs:

- **SOAP API requests** - Used for Full Load and Apply Changes replication tasks. This API will be used for Full Load when the Soap API option is selected [in the Advanced tab](#).
- **Bulk API** - Used for Full Load replication only. This API will be used for Full Load when the Bulk API option is selected [in the Advanced tab](#) (the default). In some cases, such as lookup and replication of metadata, the Soap API will be used even when Bulk API is selected.
- **Streaming API** - Used for Apply Changes replication tasks

Both Bulk API and Streaming API consumption can be monitored using Salesforce REST API requests. The output of these requests is a JSON file containing current quota consumption values. The relevant values from a Replicate perspective are as follows:

- `DailyBulkApiRequests` - The current Bulk API quota.
- `DurableStreamingApiConcurrentClients` - The current Streaming API allowed clients.
- `MonthlyPlatformEvents` - The number of captured Change Operations sent by Salesforce. To calculate the Change Operations quota per day, divide this value by 30.

9 Adding and managing source endpoints

For information about these quotas and their limits, and for information on monitoring Bulk API and Streaming API consumption, refer to the Salesforce Developer Help.

Quota consumption by Replicate:

Full Load only tasks

Full Load only tasks consume ~20 SOAP API requests per table (for meta-data and test connections). The actual number of requests consumed is proportionate to the number of tables being replicated: the greater the number of tables, the lesser the number of requests required per table.

When Bulk API is used for Full Load:

If the endpoint's [PK chunking option](#) is enabled, from the perspective of SOAP API request consumption, each chunk is the equivalent of a table.

Additionally, the endpoint consumes a single Bulk API request per table and a single Bulk API request per PK chunk (when enabled).

For each table:

$(\sim 20 * [\text{number of chunks}] \text{ SOAP API units}) + (1 + [\text{number of chunks}] \text{ Bulk API units})$

When SOAP API is used for Full Load:

For every 200 rows unloaded, one quota unit is consumed. For example, a table with 1000 rows will consume five quota units.

For each table:

$(\sim 20 * \text{SOAP API units}) + (1 + [\text{Rows of table}] / 200 \text{ SOAP API units})$

Apply Changes only tasks

Apply Changes only tasks consume 10 SOAP API requests per table. For Streaming API requests, Apply Changes only tasks consume a single quota unit per applied change. Changes that require column lookup (when there is a big textarea value) consume a single SOAP API request per change.

For Apply Changes of a single table:

$(10 \text{ SOAP API units}) + ([\text{number of changes}] \text{ Streaming API units}) + [\text{number of rows require lookup}] \text{ soap API units}$



When working with Apply Changes and/or Store Changes tasks, the Salesforce endpoint subscribes to a single Streaming API channel which logs change information for all tables. From a quota perspective, to prevent the Streaming API channel from being read multiple times (thereby increasing quota consumption), best practice is to set up a single task.

Limitations and considerations

The following limitations apply when using Salesforce as a source in a Replicate task:

- The following object types are not supported:
 - External objects
 - Big objects
 - Non-replicable objects (i.e. tables not created by the customer such as Salesforce system tables)
- Replicate performs CDC using the Salesforce CDC API and is therefore subject to its limitations. For example:
 - Salesforce CDC does not support all of the Salesforce standard objects. However, objects that are not supported using Salesforce CDC can still be replicated in a Full Load task. For a list of supported Salesforce CDC objects, visit: https://developer.salesforce.com/docs/atlas.en-us.change_data_capture.meta/change_data_capture/cdc_object_support.htm
 - UPDATES performed by the Salesforce Bulk API will not generate a Salesforce notification and will therefore not be captured.
 - The resume processing from "Source change position" Run option uses the *Salesforce* retention time.
 - As Salesforce does not generate a FULL after-image, the following limitation applies: The [INSERT the missing target record](#) option and the Batch optimized apply mode of [Apply Changes using SQL MERGE](#) option are not supported. If either of these options is selected, NULL will be inserted into the target for all corresponding source columns that were not changed in the UPDATE operation. If one of these columns is NOT NULL, the operation will result in an error. If a table contains a LOB column, and the column was not changed, the task stops with a fatal error.
- DDLs are not supported.
- The "Estimated Count" in the Full Load monitoring statistics will be N/A.
- Only the user + password + security token authentication method is supported.
- Shield Platform Encryption is not supported.
- The source_lookup Data Enrichment function is not supported. For more information on this function, see *Data Enrichment functions (page 1157)*
- The "Start from timestamp" Run option is not supported.
- The **AR_H_USER** header column is not supported. For information on using header columns, see *Headers (page 1143)*.
- Using the [Filter option](#) may impact processing time as Replicate first retrieves *all of the data* and then performs the filtering.
- Capturing DELETE and UPDATE operations is not supported when using [filters](#).
- Transformations that affect multiple columns are not supported.
- When storing changes in Change Tables, the before-image will not be stored. Only the changed columns of the after-image will be stored. All other columns will have NULL values.
- Formula fields are not supported and will not be shown as part of the table's metadata.

9 Adding and managing source endpoints




- Capture of Gap Events and Overflow Events is not supported. On encountering such events, Replicate will suspend the affected tables and require them to be reloaded. For more information on these event types, see [the Salesforce online help](#).
- During Full Load, tables that need to be queried using Replicate filters are not supported and will be suspended. The "Vote" object, for example, is not supported as it requires the data to be filtered using Parent or ID clauses.
- Replication of **IsDeleted** and **LastViewedDate** columns is not supported.
- Replication of CDP (Customer Data Platform) objects is not supported.
- Replicating data with 4-byte emoji characters is not supported.

Supported data types

The following table shows the Salesforce source data types that are supported when using Qlik Replicate and the default mapping to Qlik Replicate data types.

For information on the target data types, see the section for the target endpoint you are using. For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

Data types

Salesforce Data Types	Qlik Replicate Data Types
string	WSTRING
boolean	BOOLEAN
int	INT4
double	REAL8
date	DATE  <i>The DATE will be represented in UTC format on the target.</i>
time	TIME  <i>The TIME will be represented in UTC format on the target.</i>
datetime	DATETIME  <i>The DATETIME will be represented in UTC format on the target.</i>
ID	WSTRING
reference	WSTRING
currency	REAL8

Salesforce Data Types	Qlik Replicate Data Types
textarea	If length is <= 1000 WSTRING If length is > 1000 CLOB
percent	REAL8
phone	WSTRING
url	WSTRING
email	WSTRING
combobox	WSTRING
picklist	WSTRING
multipicklist	WSTRING

Unsupported data types

The following data types are not supported:

- ANYTYPE
- BASE64



Compound columns (e.g. **Address**) will not be replicated in their original form. Only their underlying columns (such as country, city, etc.) will be replicated (as separate columns).

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a Salesforce source endpoint to Qlik Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Salesforce database. This is optional.
4. Select **Source** as the database **role**.
5. From the **Type** drop-down list, select **Salesforce**.
6. From the **Connection Type** drop-down list, choose one of the following:

- **Production** to connect to a Salesforce production account using the default URL.
 - **Sandbox** to connect to a Salesforce sandbox account using the default URL.
 - **Custom** to connect to a Salesforce account at a custom URL.
7. Provide the following connection settings:
- **URL:**
If the selected **Connection Type** is Production, the URL will be `https://login.salesforce.com`
If the selected **Connection Type** is Sandbox, the URL will be `https://test.salesforce.com`
If the selected **Connection Type** is Custom, specify the URL of your Salesforce database.
 - **Username:** The username for accessing the Salesforce database.
 - **Password:** The password associated with the above username.
 - **Security Token:** If you are in possession of a Salesforce Security Token, you can enter it here.
8. To verify that your connection settings are correct, click **Test Connection**.
9. Click **Save** to save your settings.

Setting advanced connection properties

In the **Advanced** tab, you can enable PK chunking and set internal parameters and review your settings.

- **Use labels for column name** - Select this option if you wish to use Salesforce field labels as column names.



- *If you clear this option while the task is stopped, you will need to select the **DROP and CREATE table** option in the task settings' **Full Load Settings** tab, and then reload the task.*
- *Tables containing multiple instances of the same label are not supported.*

- **Replicate encrypted columns** - Select this option to include encrypted columns in the replication task.



- *To replicate encrypted columns, the [user specified in the General tab](#) must have the Salesforce "View Encrypted Data" permission.*
- *Encrypted columns will not be encrypted on the target.*
- *Replicating encrypted columns will not impact performance*

- **Perform Full Load using** - Choose whether to use **Bulk API** or **SOAP API** for Full Load replication. While Bulk API (the default) is much faster, it is subject to more restrictive quotas and may also sometimes be unavailable due to heavy load.
See also: *Calculating the Salesforce quota (page 468)*
- **Enable PK chunking (Available for Bulk API Full Load only)** - Select to enable Primary Key chunking of supported tables during Full Load. PK chunking can be used to split bulk queries on very large tables into chunks based on the record IDs, or primary keys of the queried records.



When this option is enabled, all selected tables will be replicated regardless of whether or not they support PK chunking. Unsupported tables will simply be replicated without being split into chunks.

As each chunk is processed as a separate batch that counts toward the user's daily batch limit, this option should not be enabled for standard-sized tables. In fact, Salesforce recommends that PK chunking should only be enabled when querying tables with more than 10 million records or when a bulk query consistently times out. Therefore, instead of enabling this option for all tables, it is recommended to set up two different tasks with two different Salesforce endpoints - one (with **Enable PK chunking** turned on) for replication of tables containing 10 million records or more, and the other (with **Enable PK chunking** turned off) for any other tables that you need to replicate. For more information on the PK chunking feature as well as a list of supported objects, see [the Salesforce online help](#).

- **Chunk size** - Specify the number of records within the ID boundaries for each chunk. The default is 200,000, and the maximum size is 250,000.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

9.27 Using Salesforce (Incremental Load) as a source

This section describes how to set up and use the Salesforce source endpoint in a replication task. The Salesforce (Incremental Load) source endpoint is similar to the Salesforce (Streaming CDC) source endpoint, although the method each uses to capture changes is different.

Salesforce (Streaming CDC) uses the Salesforce Streaming API which is [subject to several limitations](#), although it does provide better performance. Salesforce (Incremental Load), however, is based on change polling which, although slower than Salesforce (Streaming CDC), has the advantage of supporting almost all tables.

Basically, if you need to capture changes from tables that are not supported with Salesforce (Streaming CDC) then choose Salesforce (Incremental Load). Otherwise, choose Salesforce (Streaming CDC) for its better performance during change capture.

In this section:

- *Prerequisites (page 476)*
- *Calculating the Salesforce quota (page 476)*
- *Limitations and considerations (page 477)*
- *Supported data types (page 478)*
- *Setting general connection properties (page 479)*
- *Setting advanced connection properties (page 480)*

Prerequisites

The following section describes the prerequisites for replicating from a Salesforce database.

- The Salesforce endpoint supports replication of standard objects enabled for replication and enterprise custom objects.
To enable replication of custom objects, the following permissions should be set on the object's detail page:
 - Allow Sharing
 - Allow Bulk API Access - Required when the [Full Load](#) method is set to either **Bulk API** or **Select API automatically**.

Calculating the Salesforce quota

The Salesforce endpoint leverages Salesforce APIs to provide Full Load and Change Processing replication functionality. When using the Salesforce endpoint, it's important to be aware of the API quota limits in order to plan your replication tasks accordingly.

Replicate utilizes the following Salesforce APIs:

- **SOAP API requests** - Used for Full Load and Apply Changes replication tasks. This API will be used for Full Load when the Soap API option is selected [in the Advanced tab](#).
- **Bulk API 2.0** - Used for Full Load and Apply Changes replication tasks. This API will be used for Full Load when the **Bulk API** option is selected [in the Advanced tab](#) (the default). In some cases, such as lookup and replication of metadata, the Soap API will be used even when **Bulk API** is selected.

SOAP API and Bulk API 2.0 consumption can be monitored using Salesforce REST API requests (see https://developer.salesforce.com/docs/atlas.en-us.238.0.api_rest.meta/api_rest/resources_limits.htm). The results of these requests are written to a JSON file containing current quota consumption values. The relevant value from a Replicate perspective is `DailyApiRequests` which is the current SOAP API quota and `DailyBulkV2QueryJobs`, which is the current Bulk API 2.0 quota.

For information about the Bulk API quota and its limits, and for information on monitoring Bulk API 2.0 consumption, refer to the Salesforce Developer Help.

Quota consumption by Replicate:

Replicate tasks consume approximately 20 SOAP API requests per table (for metadata and test connections). The actual number of requests consumed is proportionate to the number of tables being replicated: the greater the number of tables, the lesser the number of requests required per table.



The estimations below are also true for a single CDC round. For an explanation of the term "round", see [Setting general connection properties](#) (page 479).

When Bulk API 2.0 is used

The endpoint consumes a single Bulk API 2.0 request per table and a single SOAP API request per data chunk.

For each table:

- **Bulk API 2.0:**
1 unit
- **SOAP API:**
5-20 units + [number of chunks (default chunk size is 100,000 rows)]
units

Example:

A table with 1,000,000 rows will use a single Bulk API 2.0 unit and approximately 25 SOAP API units.

When SOAP API is used

For every 200 rows unloaded, one quota unit is consumed. For example, a table with 1000 rows will consume five quota units.

For each table:

5-20 Units + (rows of table / ~200 rows) units

Limitations and considerations

The following limitations apply when using Salesforce as a source in a Replicate task:

- Object types limitations:
 - External objects are not supported as they cannot be queried.
 - Big objects are supported using Full Load replication only. If you want to replicate updates to Big Objects, you will need to schedule your task - which must be configured with Full Load replication - according to the expected update frequency.
For information on scheduling tasks, see *Scheduling jobs (page 1272)*.
- DDLs are not supported.
- The "Estimated Count" in the Full Load monitoring statistics will be N/A.
- Only the user + password + security token authentication method is supported
- Shield Platform Encryption is not supported.
- The source_lookup Data Enrichment function is not supported. For more information on this function, see *Data Enrichment functions (page 1157)*
- The **AR_H_USER** header column is not supported. For information on using header columns, see *Headers (page 1143)*.
- When storing changes in Change Tables, the before-image will not be stored.
- Replication of tables without tracking columns (for example, **lastModifiedDate**) is supported in Full Load only tasks.
- Some tables can only be replicated by using a Replicate filter.

Example:

select. * from vote is not supported

select * from vote where account_id=1234 is supported

For more information, see https://developer.salesforce.com/docs/atlas.en-us.200.0.object_reference.meta/object_reference/sforce_api_objects_vote.htm



- The transaction ID displayed in the **AR_H_TRANSACTION_ID** header column will be a Replicate-generated ID, and not the original Salesforce ID (txn_id).
- Replicating data that has been recovered from the Recycle Bin (Undelete) will be shown as an UPDATE operation in the **Change Processing** tab in the **Monitor** view. To prevent this, set the [Apply Conflicts error handling](#) for UPDATES to: **No record found for applying an UPDATE: INSERT the missing target record**.
- Compound fields are fully supported using SOAP API only. Therefore, if you need to replicate such fields, it is recommended to choose SOAP API as your Full Load/CDC method in the [Advanced tab](#). If you do choose Bulk API, the underlying fields will be replicated as separate columns instead only without the additional compound field.
- As the data is captured directly from Salesforce tables, a sequence of operations might be merged into a single operation. For example, an INSERT followed by UPDATES to the same record in the same capture cycle will be handled as an INSERT.
- Replication of CDP (Customer Data Platform) objects is not supported.
- Replicating data with 4-byte emoji characters is not supported.


Supported data types

The following table shows the Salesforce source data types that are supported when using Qlik Replicate and the default mapping to Qlik Replicate data types.

For information on the target data types, see the section for the target endpoint you are using. For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

Data types

Salesforce Data Types	Qlik Replicate Data Types
string	WSTRING
boolean	BOOLEAN
int	INT4
double	REAL8
date	DATE
	 <i>The DATE will be represented in UTC format on the target.</i>
time	TIME
	 <i>The TIME will be represented in UTC format on the target.</i>

Salesforce Data Types	Qlik Replicate Data Types
datetime	DATETIME  The DATETIME will be represented in UTC format on the target.
ID	WSTRING
reference	WSTRING
currency	REAL8
textarea	If length is <= 1000 WSTRING If length is > 1000 CLOB
percent	REAL8
phone	WSTRING
url	WSTRING
email	WSTRING
combobox	WSTRING
picklist	WSTRING
multipicklist	WSTRING

Unsupported data types

The following data types are not supported:

- ANYTYPE
- BASE64

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a Salesforce source endpoint to Qlik Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.

3. In the **Description** field, type a description that helps to identify the Salesforce database. This is optional.
4. Select **Source** as the database **role**.
5. From the **Type** drop-down list, select **Salesforce (Incremental Load)**.
6. From the **Connection Type** drop-down list, choose one of the following:
 - **Production** to connect to a Salesforce production account using the default URL.
 - **Sandbox** to connect to a Salesforce sandbox account using the default URL.
 - **Custom** to connect to a Salesforce account at a custom URL.
7. Provide the following connection settings:
 - **URL:**
If the selected **Connection Type** is **Production**, the URL will be `https://login.salesforce.com`
If the selected **Connection Type** is **Sandbox**, the URL will be `https://test.salesforce.com`
If the selected **Connection Type** is **Custom**, specify the URL of your Salesforce database.
 - **Username:** The username for accessing the Salesforce database.
 - **Password:** The password associated with the above username.
 - **Security Token:** If you are in possession of a Salesforce Security Token, you can enter it here.
8. To verify that your connection settings are correct, click **Test Connection**.
9. Click **Save** to save your settings.

Setting advanced connection properties

In the **Advanced** tab, additional configuration options are available, including fine-tuning performance, setting up proxy server connectivity, and setting internal parameters.

Data loading

- Select one of the following **Check for changes** options:
 - **Periodically** - When this option is selected, Replicate checks for changes according to the **Incremental load interval (seconds)**
- **On initial run and then as scheduled** - When this option is selected, to continue capturing changes after the initial task run, you need to schedule the task to run according to your needs.



The interval will start after each "round". A round can be defined as the time it takes for Replicate to read the changes from the source tables, and send them to the target (as a single transaction). The length of a round varies according to the number of tables and changes. So if you specify an interval of 10 minutes and a round takes 4 minutes, then the actual time between checking for changes will be 14 minutes.



This option is only relevant if the interval between the CDC cycles is 24 hours or more.

For information on scheduling tasks, see *Scheduling jobs (page 1272)*.

- **Full Load method:** Choose **Select API automatically** (default), **SOAP API**, or **Bulk API**.

9 Adding and managing source endpoints

While Bulk API is much faster, it is subject to more restrictive quotas and may also sometimes be unavailable due to heavy load.

See also: *Calculating the Salesforce quota (page 476)*

- **API switchover threshold (records):** Only relevant when **Select API automatically** is selected. If the number of records per table matches or exceeds the specified value, Replicate will retrieve the data using the Bulk API (if the object support Bulk API); otherwise, it will retrieve the data using the SOAP API.
- **CDC method:** Choose **Select API automatically** (default), **SOAP API**, or **Bulk API**. While SOAP API is slower than Bulk API, it has less restrictive quotas and, unlike Bulk API, is usually always available (even during heavy load).
See also: *Calculating the Salesforce quota (page 476)*
 - **API switchover threshold (records):** Only relevant when **Select API automatically** is selected. If the number of records per table matches or exceeds the specified value, Replicate will retrieve the data using the Bulk API (if the object support Bulk API); otherwise, it will retrieve the data using the SOAP API.
- **Maximum data chunk size (records):** If you encounter performance issues, increasing this value might improve performance. However, excessive chunk sizes might result in connection timeouts. In such a case, reduce the value until normal operation is restored.



Only relevant when the Full Load method or CDC method is Bulk API.

- **Transaction consistency timeout (sec.):** To maintain transactional consistency, increase this value as necessary if your longest transaction exceeds 10 seconds.



Only relevant when the task replication option is set to **Full Load** and **Apply Changes**.

- **Replicate encrypted columns** - Select this option to include encrypted columns in the replication task.



- To replicate encrypted columns, the [user specified in the General tab](#) must have the Salesforce "View Encrypted Data" permission.
- Encrypted columns will not be encrypted on the target.

Metadata

Select the **Use labels for column name** check box if you want to use Salesforce field labels as column names.



- If you clear this option while the task is stopped, you will need to select the **DROP and CREATE table** option in the task settings' **Full Load Settings** tab, and then reload the task.
- Tables containing multiple instances of the same label are not supported.

Proxy server

If your Salesforce instance needs to be accessed via a proxy server, specify the access settings below.

- **Use proxy server:** Select this option to access Salesforce via a proxy server.
- **Host name:** The host name of the proxy server.
- **Port:** The port via which to access the proxy server.
- **User name:** The user name for accessing the proxy server.
- **Password:** The password for accessing the proxy server.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

9.28 Using SAP Application as a source

This section describes how to define a SAP Application as a source endpoint in a replication task.

In this section:

- *Prerequisites (page 484)*
- *Limitations and considerations (page 491)*
- *SAP Application source data types (page 491)*
- *Setting general connection properties (page 493)*
- *Setting advanced properties (page 496)*

Prerequisites

The following section describes the prerequisites for working with the SAP Application endpoint.

- *Supported SAP packages (page 484)*
- *Set up a source endpoint for your SAP application (page 484)*
- *Install the SAP NetWeaver RFC Client (page 484)*
- *Install the Replicate for SAP Client on the SAP machine (page 500)*
- *Managing business groups and tables (page 489)*
- *Target collation (page 491)*

Supported SAP packages

- Primarily SAP ERP / ECC 6.0 + all EhP levels
- All modules are supported.



When replicating from the HR module, only transparent tables are supported.

- Also supports CRM, SRM, GTS and MDG SAP Applications

See also: *Set up a source endpoint for your SAP application (page 484)*

Set up a source endpoint for your SAP application

Before you can configure the Qlik Replicate SAP endpoint, you first need to configure one of the following source endpoints, according to your SAP Package type:

- Microsoft SQL Server
See also: *Using Microsoft SQL Server as a source (page 348)*
- Oracle
See also: *Using Oracle as a source (page 416)*
- IBM DB2 for LUW
See also: *Using IBM DB2 for LUW as a source (page 267)*
- SAP HANA



Change capture is supported using Trigger-based CDC only

See also: *Using SAP HANA as a source (page 537)*

Install the SAP NetWeaver RFC Client

This topic explains how to copy the required SAP NetWeaver RFC Client files to the Qlik Replicate machine.



Replicate has been certified to work with NetWeaver RFC SDK 7.50 Patch 1021, which can be downloaded from the SAP Service Marketplace. As newer NetWeaver RFC SDK versions are constantly being released, it is not possible to certify compatibility with every version. We therefore recommend working with the certified version, even though later versions might also work without any issues.

Windows


Extract the contents of the *NWRFC_XXX.SAR* file and then copy the *.dll* files from the *nwrfcsdk/lib* directory to the Qlik Replicate bin directory.

Linux

Extract the contents of the *NWRFC_XXX.SAR* file and then copy the *.so* files from the *nwrfcsdk/lib* directory to the Replicate **lib** directory.

Install the Replicate for SAP Client on the SAP machine

This section describes how to install the transports that make up the Qlik Replicate for SAP Client (R4SAP).

1. Go to  [Product Downloads](#).
2. Select **Qlik Data Integration**.
3. Scroll down the **Product** list and select Replicate.
4. In the **Download Link** column, locate the **QlikReplicate_<version-build>_R4SAP.zip** file. Before starting the download, check the **Version** column to make sure that the R4SAP version you are downloading corresponds to your current Replicate version.

There are seven transports in total, which are provided in the following folders:

- DeleteCode - Required for uninstalling the Replicate for SAP Client.
- DeletePackage - Required for uninstalling the Replicate for SAP Client.
- DeleteCodeCommon - Required for uninstalling the Replicate for SAP Client.
- DeletePackageCommon - Required for uninstalling the Replicate for SAP Client.
- InstallCode (e.g. K902086.ESD) - The main transport.
- InstallCodeECC (e.g. K901271.ESD) - A transport with additional logic for ECC SAP systems .
- InstallConfig (e.g. K900012.R4S) - The configuration transport.

The installation procedure



- *The transports must be installed in the order they appear below. Installing the transports in the incorrect order or omitting/skipping any of the transports will result in unexpected issues with replication tasks.*
- *When installing the Replicate for SAP client, do not chose the "overwrite" option if Gold Client is already installed on the Replicate server machine. Otherwise, the Gold Client software will stop working.*

To install the transports on ECC systems:

1. Install the main transport file (InstallCode).
2. Install the ECC-specific transport (InstallCodeECC).
3. Install the configuration transport (InstallConfig).

To install the transports on non-ECC systems (e.g. CRM):

1. Install the main transport file (InstallCode).
2. Install the configuration transport (InstallConfig).



If you are [applying a patch or upgrading](#) the Replicate for SAP Client, you should only install the main transport and the ECC-specific transport (if upgrading on an ECC system). Do not install the configuration transport again, or any customizations made to the configuration will be lost.

Permissions required for installing Replicate for SAP Client

Replicate for SAP delivers its own authorization object: ZR4SAP. In addition to this authorization object, there are additional authorizations that need to be enabled for the Replicate software.

SAP users for Replicate

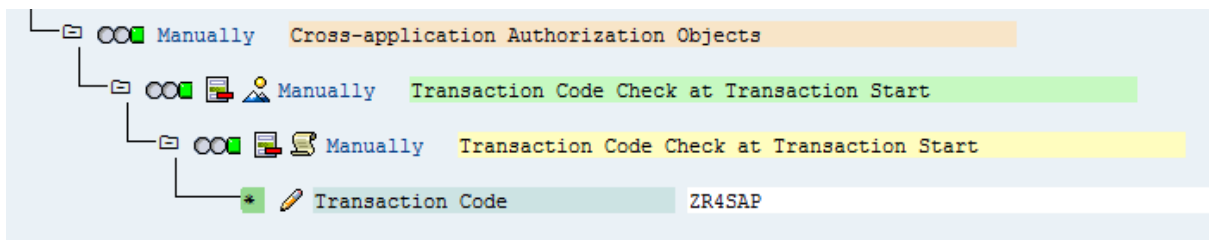
A dialog user in SAP is required to access the Replicate for SAP Client GUI in SAP. In addition, a communication user is required to support the RFC calls from the Replicate software to the SAP system.

Identify existing users in SAP or create dedicated users for the Replicate software.

Authorizations for Replicate

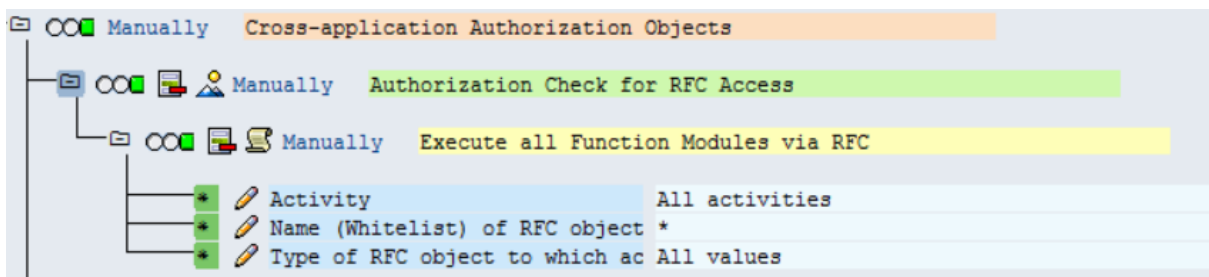
Both the dialog and communication users will need to be assigned to a role with authorization object S_TCODE and value ZR4SAP.

Authorization Object S_TCODE with value ZR4SAP:

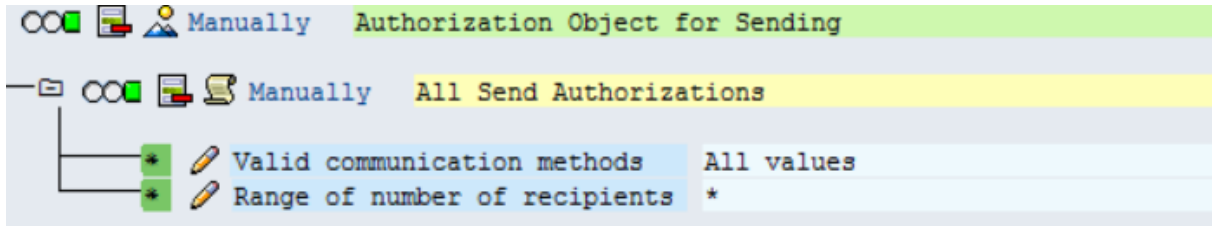


The communication user will also require the following authorization objects: S_RFC and S_OC_SEND.

Authorization Object S_RFC:



Authorization Object S_OC_SEND:



Importing the Replicate transports into the SAP system

There are two types of files required to import the ABAP objects into the SAP system: the data-file and the co-file.

Importing the data-file

The data-file begins with an "R"

The data-file should be placed in the `/usr/sap/trans/data` file system or in the directory of the server where the transports are stored.

- Typically this is a shared directory, but if not, individually place the file into that directory location for all SAP host servers.
- This file must be accessible by all systems where the Replicate for SAP Client is to be installed.
- Set the permissions on the file to A11 for the user, Read and Execute for the group, and Read and Execute for others.
- The owner of the file should be the `<sid>adm` user of the system to be installed. The group ownership should be `sapsys`.

Importing the co-file

The co-file begins with a "K"

The co-file should be placed in the `/usr/sap/trans/cofiles` file system or in the directory of the server where the transports are stored.

- Typically this is a shared directory, but if not, individually place the file into that directory location for all SAP host servers.
- This file must be accessible by all systems where the Replicate for SAP Client is to be installed.
- Set the permissions on the file to A11 for the user, Read and Execute for the group, and Read and Execute for others.
- The owner of the file should be the `<sid>adm` user of the system to be installed. The group ownership should be `sapsys`.

Once the files are in the correct location, import the transport into the system using either the Operating System level transport tools (TP), or the Transport Management System (TMS) internally within SAP.

Importing the transports via TP

1. Log on to the system at the Operating System level as the `<sid> adm`.
2. Change the directory to `/usr/sap/trans`
3. Add the transport to the R/3 buffer with the following command:

```
# tp 'addtobuffer SID'
```

4. Import the transport to the target R/3 system with the following command:

```
# tp 'import SID client=000 U16'
```

The expected result of the `addtobuffer` step is a successful return code of ``0'`.

If problems occur during the `addtobuffer` step, it is likely there is a problem with the files. They may be missing, in the wrong location, or have incorrect ownership or permissions.

The expected result of the `import` step is a successful return code of either ``0'` or ``4'`. A return code of ``8'`, ``12'` or ``16'` indicates transport failure. Return codes higher than ``16'` indicate a major failure within the transport tool. If this occurs, check the present working directory to ensure the correct location. Also, check the files for existence, location, and proper ownership and access.

If problems exist during the import, retry the import step. If problems persist, check the import and activation logs for failure reason. These files are in the `/usr/sap/trans/log` location and named.R6U (the ``?'` stands in as a wildcard).

Importing the transports via TMS

Starting from R/3 version 4.0, SAP allows importing transports through the SAP system via transaction code STMS.



Security authorization in the SAP system must include proper access to import the transport request.

1. Sign-on to SAP system with a User ID that contains proper authority.
2. Execute transaction STMS.
3. Select the **Transport** (Truck) icon from the toolbar.
4. Select the desired system for import.
5. Add the Transport to the import queue by selecting the following path from the menu list:
Extras > Other Requests > Add
Add the transport request number to the proper import queue and execute. Re-authentication of the user's SAP User ID is likely in order to complete the step.
If an Information message is received that the "Transport request is invalid" check that the transport number was typed correctly. Otherwise, it may indicate a problem with the files. Verification of existence, location, permissions, or ownership may be needed.
6. Import the Transport request by selecting the transport number from the queue, and clicking the **Import** (Truck) icon from the toolbar. Set the target client to either ``000'` or any other valid client within the system and execute with the truck icon. Once again, re-authentication of the SAP User ID may be necessary. The transport will execute in asynchronous mode; a record of success or failure can be found in the transport logs.
7. The system will return to the import queue screen, where the Transport results can be checked. Select the **Logs** icon from the toolbar, or follow the menu path:
Request > Display > Logs
Locate the source system and verify all relevant logs. For this transport there should be 5 logs:
 - **DD Import**
 - **DD Activation**
 - **Import**

- **Check Versions**
- **ABAP/scrn. Generation**

All logs should display Ended OK (return code 0) or Ended with warning (return code 4). If any logs are missing, or display a return code of 8 or higher, follow the instructions in step 6 to re-import the transport.

Upgrading/patching and uninstalling the Replicate SAP Client

The following instructions explain how to upgrade/patch, and uninstall the Replicate SAP Client.



If you are applying a patch or upgrading the Replicate for SAP Client, you should only install the main transport and the ECC-specific transport (if upgrading on an ECC system). Do not install the configuration transport again, or any customizations made to the configuration will be lost.

To upgrade or patch the Replicate SAP Client on ECC Systems

1. Apply the new main transport (InstallCode.zip).
2. Apply the new ECC-specific transport (InstallCodeECC.zip).

To upgrade or patch the Replicate SAP Client on non-ECC systems

- Apply the new main transport (InstallCode.zip).

To uninstall the Replicate SAP Client

1. Apply the "Delete Main" transport (DeleteCode.zip).
2. Apply the "Delete Package" transport (DeletePackage.zip).



Do not apply items 3 and 4 if you are also running Qlik Gold Client Solution.

3. Apply the "Delete Common" transport (DeleteCodeCommon.zip)
4. Apply the "Delete Common Package" transport (DeletePackageCommon.zip)

Managing business groups and tables

This prerequisite is only necessary if you want to edit the default Business Groups and/or tables before replicating them.

Before you can manage business groups and tables, you first need to launch the SAP Client UI.

To launch the SAP Client UI:

1. Open your SAP client console.
2. Double-click one of the SAP Application Sources.
You will be prompted for your user name and password.
3. Enter your credentials for logging in to the selected SAP Application Source.
4. Enter /nzsap in the drop-down list at the top of the console and then press [Enter].

5. Click the **Business Groups Configuration** button.
A list of Business Groups is displayed in the left pane.

Managing business groups

To add a new Business Group:

1. Click the **Create** toolbar button.
The **Create Business Group** dialog box opens.
2. Enter a name for your Business Group and then press [Enter].
The new Business Group is added to the **Business Groups** list in the left pane.

To duplicate a Business Group




1. Click the **Copy** toolbar button.
The **Business Group Configuration** dialog box opens.
2. In the **New Bus Object** field, enter a name for the new Business Group and then press [Enter].
The duplicated Business Group is added to the **Business Groups** list in the left pane.

To delete a Business Group




- Select the Business Group you want to delete and then click the **Delete** toolbar button.
The Business Group is deleted.

Managing tables

To add a new table to a Business Group:

1. In the left pane, expand the desired Business Group.
2. Double-click the **Tables** icon.
A list of tables is shown in the right pane.
3. Click the  button above the table list to enter Edit mode.
4. Click the  button that appears to the right of the  button.
An empty row is added to the tables list.
5. Enter the **Table Name** (i.e. the virtual ABAP table) and the name of the corresponding **Source Table** (i.e. the physical table).
6. To save your changes click the **Save** button in the main toolbar.

To remove a table from a Business Group:

1. In the left pane, expand the desired Business Group.
2. Double-click the **Tables** icon.
A list of tables is shown in the right pane.
3. Click the  button above the table list to enter Edit mode.
4. Select the table you want to delete.
5. Click the  button that appears to the right of the  button.

The table is deleted.

6. To save your changes click the **Save** button in the main toolbar.

Accessing tables with a non-English language

To enable replication of non-English SAP tables, add them to the configuration as follows:

1. From the zr4sap transaction, click the **Metadata Language Configuration** button.
2. Click the **Add** button and select the appropriate language.
3. Click **Save**.

To remove a previously configured language, select the desired row and click the **Delete** button.

Target collation

As SAP is case-sensitive, when a replication task is defined with a SAP Application source, the target database needs to be set up with case-sensitive collation.

Limitations and considerations

When using SAP Application as a source endpoint in a Replicate task, the following limitations apply.

- During Full Load of clustered and pooled tables, the **Estimated Count** value in Monitor view may be incorrect. Additionally, as the Progress Bar is based on the **Estimated Count** value, it may not represent an accurate portrayal of the actual task progress.
- Replication of views is not supported.
- When using the HR module as a source, only transparent tables are supported.
- Replicating data with 4-byte emoji characters is not supported.
- A task with a SAP Application source and a File Channel target may replicate some tables twice - the requested table and the underlying table. To prevent this from happening, exclude the underlying table from the list of tables to be replicated.
- When a task is defined with a SAP Application source, the **Applied Changes Details** monitoring metrics in the [Change Processing](#) tab may be incorrect for clustered and pooled tables.
- Changes are applied to cluster tables in transactional mode only. If you need the [Batch optimized apply](#) functionality, consider using the SAP Application (DB) endpoint instead.
- [Unlimited LOB mode](#) is not supported.
- The source_lookup Data Enrichment function is not supported. For more information on this function, see *Data Enrichment functions (page 1157)*

SAP Application source data types

The SAP Application endpoint for Qlik Replicate supports most SAP data types. The table below shows the SAP source data types that are supported when using Qlik Replicate and the default mapping to the Qlik Replicate data types.

For information on how the data type is mapped to the target, see the chapter for the target endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

9 Adding and managing source endpoints

Data types

ABAP TYPE	ABAP Type Description	SAP Type	Qlik Replicate Data Type
h	Table type	-	BYTES
V	Character string (old Dictionary type VARC)	-	STRING
C	Character string	-	STRING
N	Character string with only digits	-	STRING
D	Date (string: YYYYMMDD)	-	DATE
T	Time (string: HHMMSS)	-	TIME
X	Byte sequence	INT4 (4-byte integer)	I4
		INT2 (2-byte integer)	I2
		INT1 (1-byte integer)	I1
		ELSE	If backend type is NUMERIC: NUMERIC If length = 0: BLOB If length > 0: BYTES
I	Integer number (4-byte integer with sign)	-	INT4
b	2-byte integer	-	INT2
s	1-byte integer	-	INT1
P	Packed number	-	NUMERIC
F	Floating point number to accuracy of 8 bytes	-	R8
g	Character string with variable length	-	STRING
y	Byte sequence with variable length	-	BLOB
			BYTES
u	Structured type, flat	-	BYTES

ABAP TYPE	ABAP Type Description	SAP Type	Qlik Replicate Data Type
v	Structured type, deep	-	BYTES
r	Reference to class/interface	-	BYTES
i	Reference to data object	-	BYTES
n	Numeric text	NUMC	STRING

Setting general connection properties

This section describes how to set up connection parameters for a specific [SAP Application server](#) or for an [SAP system using load balancing](#).

To connect to a specific SAP Application server

1. In the Qlik Replicate console, click the **Manage Endpoint Connections** toolbar button to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button.
2. In the **Name** field, enter a display name for your endpoint.
3. Optionally, in the **Description** field, enter a description for the SAP Application endpoint.
4. Select **Source** as the database **role**.
5. Select **SAP Application** as the database **Type**.



*When this endpoint is used as a duplicated source in a Log Stream Staging setup, select the **Read changes from log stream** check box and then select the relevant Log Stream Staging task from the drop-down list. For information on setting up and managing Log Stream Staging tasks, see [Using the Log Stream \(page 1072\)](#).*

6. From the **Connection type** drop-down list, select **Type A**.
7. In the **Server name** field, enter the IP address of the Application Server on which the SAP Application source is located.
8. In the **Instance number** field, enter the instance number of the SAP Application source you want to replicate.
9. In the **Client** field, enter the client number of the SAP Application source you want to replicate.
10. Enter your credentials (**User Name**, **Password**) for accessing the SAP Application source.



These are the credentials for the [communication user created earlier in SAP](#).

11. In the **Backend endpoint** field, click the Browse button and then select the name of the Qlik Replicate endpoint you configured earlier. See also [Set up a source endpoint for your SAP application \(page 484\)](#).

To connect to an SAP system using load balancing

1. In the Qlik Replicate console, click the **Manage Endpoint Connections** toolbar button to open the **Manage EndpointsConnections** dialog box.

2. In the **Name** field, enter a display name for your endpoint.
3. Optionally, in the **Description** field, enter a description for the SAP Application endpoint.
4. Select **Source** as the database **role**.
5. Select **SAP Application** as the database **Type**.
6. From the **Connection type** drop-down list, select **Type B**.
7. In the **Message server** field, enter the host name or IP address of the message server host.
8. In the **Application servers group name** field, enter the name of the SAP server group. This is an optional group of application servers in a load balancing connection.
9. In the **SAP system name** field, enter the SAP R/3 name.
10. In the **Message server service** field, enter the name of the SAP message server service as specified in the following file:
<system drive>:\WINDOWS\system32\drivers\etc\services
If you do not specify a value, the Data Provider for SAP uses the following default name:
sapms<R/3 system name>
11. In the **Client** field, enter the System ID of the SAP Application source you want to replicate.
12. Enter your credentials (**User Name, Password**) for accessing the SAP Application source.



These are the credentials for the [communication user created earlier in SAP](#).

13. In the **Backend endpoint** field, click the Browse button and then select the name of the Qlik Replicate endpoint you configured earlier. See also *Set up a source endpoint for your SAP application (page 484)*.

Security

In the **Security** settings, you can configure Secure Network Communication (SNC).

Prerequisites for working with SNC

Follow the steps below to install the Secure Network Communication (SNC) client on the Replicate machine.

What you need:

- An exported certificate (.crt) of the SAP server
- SAPCAR.EXE
- SAP user (authorized customer)
- The version of the crypto library which is installed on the corresponding SAP server

Installing the SNC client

1. Create a workspace folder for the SAP SNC files and binaries (hereafter referred to as "your SNC folder"), for example: "C:\snc\"
2. Copy the exported server certificate and **SAPCAR.EXE** to your SNC folder.
3. Go to <https://support.sap.com/en/my-support/software-downloads.html> and search for **SAPCRYPTOLIB** under **Installations & Upgrades**. Download the 64-bit **.SAR** to your SNC folder.
4. Open a command prompt and change the working directory to your SNC folder. Then run the following command to unpack the content of the **.SAR** to your SNC folder:
sapcar -xvf LibName.sar

Example:

```
sapcar -xvf SAPCRYPTOLIBP_8541-20011731_32.SAR
```

5. Add system environment variables as follows:
 - a. Add a system environment called **SECUDIR** with the path to your SNC folder as its value.
 - b. Add a system environment variable called **QLIK_SNC_LIB** with the path to the **sapcrypto.dll** file as its value.
 - c. Add the newly added environment variables to the "PATH" environment variable.
6. Determine the **<PSE_File_Name>** and choose a **<PSE_PIN>** to protect it. You will need to provide this information in the next steps.

Example:

```
pseName: "CN=USR,OU=SAP,O=Qlik,C=IS" password: password123
```

7. Determine the **<SNC_NAME>**. It should look something like this: CN=USR, OU=SAP, O=Qlik, C=IS
See also *Determining the server SNC name (page 496)* below.
8. Make sure you have the required permissions to access and execute the files in the **SECUDIR** folder, and then run the following command to generate the PSE file:

```
sapgenpse get_pse -p <PSE_File_Name>.pse -x <PSE_PIN> <SNC_NAME>
```

Example:

```
sapgenpse get_pse -p usr.pse -x password123 "CN=USR,OU=SAP,O=Qlik,C=IS"
```

9. Bind the PSE file with the OS user and create the **CRED_V2** file in **SECUDIR** folder as follows:
 - a. Make a note of the OS user under which Replicate is running:
 - Windows: Open the Services console and double-click the Qlik Replicate Server service to open the properties dialog. Look in the **Log On** tab.
 - Linux: Run the `ps aux` command.
 - b. Run the following command:

```
sapgenpse seclogin -p <PSE_File_Name>.pse -x <PSE_PIN> -O <OS_USER>
```

Example:

```
sapgenpse seclogin -p usr.pse -x password123 -O SYSTEM
```

10. Generate the CRT file by executing the following command:

```
sapgenpse export_own_cert -o <PSE_File_Name>.crt -p <PSE_File_Name>.pse -x <PSE_PIN>
```

Example:

```
sapgenpse export_own_cert -o usr.crt -p usr.pse -x password123
```
11. Import the SAP Application Server Certificate (**<SERVER_CERT>**) to the PSE by executing the following command:

```
sapgenpse maintain_pk -a <SERVER_CERT>.crt -p <PSE_File_Name>.pse -x <PSE_PIN>
```

Example:


```
sapgenpse maintain_pk -a sapsys.crt -p usr.pse -x password123
```
12. To verify that the DN of the SAP Server's PSE was imported into the client, run the following command and then check the "subject" value:


```
sapgenpse maintain_pk -v -l -p <PSE_File_Name>.pse
```

Example:

```
sapgenpse maintain_pk -v -l -p usr.pse
```

Importing the client certificate

1. Connect to the SAP Application Server and navigate to the "STRUST" transaction using an authorized user.
2. Double-click the **SNC (SAPCryptolib)** folder.
3. Click  to switch to Change view.

4. Click  to import the certificate.
5. In the new dialog, enter the path to the .crt file that was created earlier, then click continue.
6. Verify the details of the certificate in the **Certificate** section.
7. Click **Add to Certificate List** to add the certificate to the list.
8. Save the changes.

Determining the server SNC name

There are two ways you can determine the server name:

- **Method 1:** Decrypt the server CRT file using the OpenSSL command. The server name will be part of the subject.
- **Method 2:** This method requires appropriate permissions. While connected to the system:
 1. Run the RZ10 transaction.
 2. Select the system profile.
 3. Select the **Extended Maintenance** option and then click **Display**.
 4. The value of the `snc/identity/as` parameter should be the SNC name.

Connection settings

Configure the SNC settings in the SAP Application endpoint as follows:

- **Activate Secure Network Communication:** Select to turn on SNC.
- **SNC name:** The SNC partner name.
Example:
`p:CN=SYS, OU=SAP, O=Qlik, C=IS`
- **SNC quality of protection** - Select one of the following:
 - **Authentication only:** Select to verify the identity of the SAP Application machine. This is the minimum protection level offered by SNC.
 - **Integrity protection:** Select to detect any changes or manipulation of the data, which might have occurred between the Replicate machine and the SAP Application machine.
 - **Privacy protection:** Select to encrypt the messages being transferred to prevent eavesdropping. Privacy protection also includes integrity protection. This is the maximum level of protection provided by SNC.
 - **Maximum security available:** The maximum level of data protection supported by the SAP Application machine.

Setting advanced properties

In the **Advanced** tab, you can set the following parameters:

- **RFC call batch:** The number of concurrent RFC calls made from Replicate back to the SAP system. If you encounter performance issues, increasing this number may help, but may also adversely affect monitoring updates.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

9.29 Using SAP Application (DB) as a source

This section describes how to define SAP Application (DB) as a source endpoint in a replication task.

You can use the SAP Application (DB) endpoint to replicate from a SAP Application with one of the [supported backend endpoints](#).

In this section:

- *Prerequisites (page 499)*
- *Limitations and considerations (page 506)*
- *SAP Application (DB) source data types (page 507)*
- *Setting general connection properties (page 508)*
- *Setting advanced properties (page 509)*

Prerequisites

The following section describes the prerequisites for working with the SAP Application (DB) endpoint.

- *Supported SAP packages (page 499)*
- *Set up a source endpoint for SAP Application (DB) (page 499)*
- *Install the Replicate for SAP Client on the SAP machine (page 500)*
- *Managing business groups and tables (page 505)*
- *Target collation (page 506)*

Supported SAP packages

- Primarily SAP ERP / ECC 6.0 + all EhP levels
- All modules are supported.



When replicating from the HR module, only transparent tables are supported.

- Also supports CRM, SRM, GTS and MDG SAP Applications

See also:

- *Set up a source endpoint for SAP Application (DB) (page 499)*

Set up a source endpoint for SAP Application (DB)

Before you can configure the SAP Application (DB) endpoint, you first need to configure one of the following backend source endpoints, according to your SAP Package type:



Replicating from an STXL table is not supported. If you need to replicate from an STXL table, use the [SAP Extractor endpoint](#) instead.

- Oracle



When using Oracle as the backend database, any tables you want to capture must be defined with ALL COLUMN SUPPLEMENTAL LOGGING.

See also: *Using Oracle as a source (page 416)*

- Microsoft SQL Server



When using Microsoft SQL Server as the backend database, transparent table fields defined as STRING or BINARY STRING are not supported.

See also: *Using Microsoft SQL Server as a source (page 348)*

- IBM DB2 for LUW

See also: *Using IBM DB2 for LUW as a source (page 267)*

- IBM DB2 for z/OS
See also: *Using IBM DB2 for z/OS as a source (page 276)*
- SAP HANA



When using SAP HANA as the backend database, the following limitations apply when the SAP HANA endpoint is configured to use Log-based CDC mode:

- Replicate creates a `hana__row_id` column on the target as a Unique Index and changes the original Primary Key/Unique Index to a standard column.
- Table DDLs are not supported and will be ignored
- Replication of transparent tables only is supported
- The Apply Conflicts error-handling policy for **No record found for applying DELETE** and **No record found for applying an UPDATE** must be set to **Ignore record**.

See also: *Using SAP HANA as a source (page 537)*

For information about general limitations when using the SAP Application (DB) endpoint, see *Limitations and considerations (page 506)*.


Backend Source Endpoint Permissions

The user specified in the underlying source endpoint settings must have SELECT permission on the following objects:

- DD02L
- DD03L
- DD02T
- DD03T
- T000
- /HTG/GC_HEAD
- /HTG/GC_TABLES1
- /HTG/GC4R_BUSGRP
- /HTG/GC4R_TABLES

Install the Replicate for SAP Client on the SAP machine

This section describes how to install the transports that make up the Qlik Replicate for SAP Client (R4SAP).

1. Go to  [Product Downloads](#).
2. Select **Qlik Data Integration**.
3. Scroll down the **Product** list and select Replicate.
4. In the **Download Link** column, locate the **QlikReplicate_<version-build>_R4SAP.zip** file. Before starting the download, check the **Version** column to make sure that the R4SAP version you are downloading corresponds to your current Replicate version.

There are seven transports in total, which are provided in the following folders:

9 Adding and managing source endpoints

- DeleteCode - Required for uninstalling the Replicate for SAP Client.
- DeletePackage - Required for uninstalling the Replicate for SAP Client.
- DeleteCodeCommon - Required for uninstalling the Replicate for SAP Client.
- DeletePackageCommon - Required for uninstalling the Replicate for SAP Client.
- InstallCode (e.g. K902086.ESD) - The main transport.
- InstallCodeECC (e.g. K901271.ESD) - A transport with additional logic for ECC SAP systems .
- InstallConfig (e.g. K900012.R4S) - The configuration transport.

The installation procedure



- *The transports must be installed in the order they appear below. Installing the transports in the incorrect order or omitting/skipping any of the transports will result in unexpected issues with replication tasks.*
- *When installing the Replicate for SAP client, do not chose the "overwrite" option if Gold Client is already installed on the Replicate server machine. Otherwise, the Gold Client software will stop working.*

To install the transports on ECC systems:

1. Install the main transport file (InstallCode).
2. Install the ECC-specific transport (InstallCodeECC).
3. Install the configuration transport (InstallConfig).

To install the transports on non-ECC systems (e.g. CRM):

1. Install the main transport file (InstallCode).
2. Install the configuration transport (InstallConfig).



If you are [applying a patch or upgrading](#) the Replicate for SAP Client, you should only install the main transport and the ECC-specific transport (if upgrading on an ECC system). Do not install the configuration transport again, or any customizations made to the configuration will be lost.

Permissions required for installing Replicate for SAP Client

Replicate for SAP delivers its own authorization object: ZR4SAP. In addition to this authorization object, there are additional authorizations that need to be enabled for the Replicate software.

SAP users for Replicate

A dialog user in SAP is required to access the Replicate for SAP Client GUI in SAP. In addition, a communication user is required to support the RFC calls from the Replicate software to the SAP system.

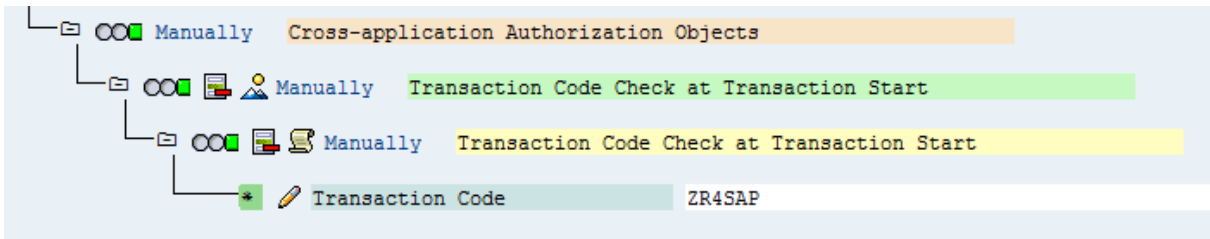
Identify existing users in SAP or create dedicated users for the Replicate software.

Authorizations for Replicate

Both the dialog and communication users will need to be assigned to a role with authorization object S_TCODE and value ZR4SAP.

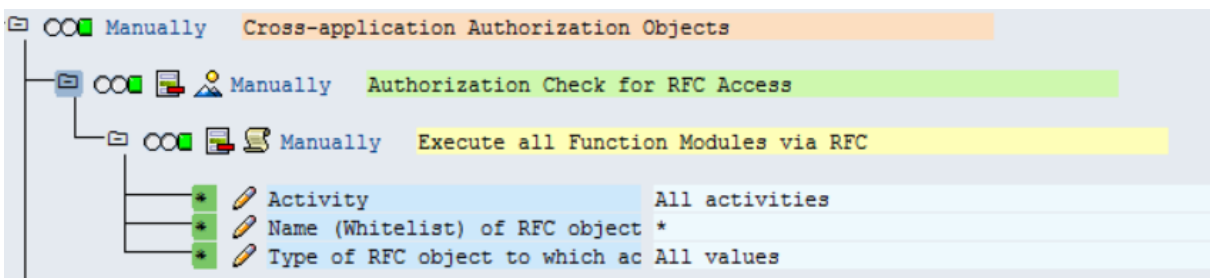
9 Adding and managing source endpoints

Authorization Object S_TCODE with value ZR4SAP:

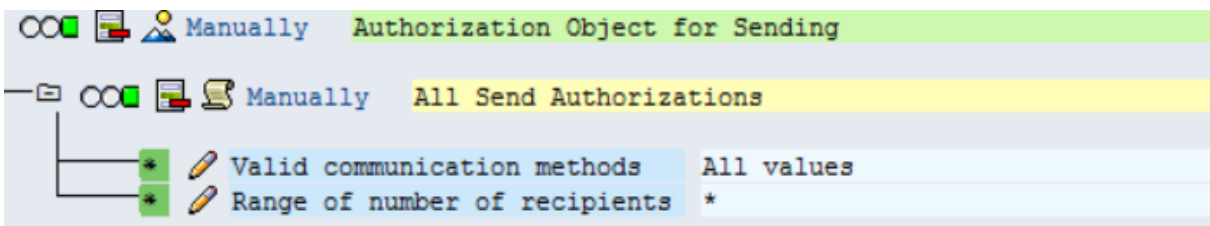


The communication user will also require the following authorization objects: S_RFC and S_OC_SEND.

Authorization Object S_RFC:



Authorization Object S_OC_SEND:



Importing the Replicate transports into the SAP system

There are two types of files required to import the ABAP objects into the SAP system: the data-file and the co-file.

Importing the data-file

The data-file begins with an "R"

The data-file should be placed in the `/usr/sap/trans/data` file system or in the directory of the server where the transports are stored.

- Typically this is a shared directory, but if not, individually place the file into that directory location for all SAP host servers.
- This file must be accessible by all systems where the Replicate for SAP Client is to be installed.
- Set the permissions on the file to A11 for the user, Read and Execute for the group, and Read and Execute for others.
- The owner of the file should be the `<sid>adm` user of the system to be installed. The group ownership should be `sapsys`.

Importing the co-file

The co-file begins with a "K"

The co-file should be placed in the `/usr/sap/trans/cofiles` file system or in the directory of the server where the transports are stored.

- Typically this is a shared directory, but if not, individually place the file into that directory location for all SAP host servers.
- This file must be accessible by all systems where the Replicate for SAP Client is to be installed.
- Set the permissions on the file to `A11` for the user, `Read` and `Execute` for the group, and `Read` and `Execute` for others.
- The owner of the file should be the `<sid>adm` user of the system to be installed. The group ownership should be `sapsys`.

Once the files are in the correct location, import the transport into the system using either the Operating System level transport tools (TP), or the Transport Management System (TMS) internally within SAP.

Importing the transports via TP

1. Log on to the system at the Operating System level as the `<sid> adm`.
2. Change the directory to `/usr/sap/trans`
3. Add the transport to the R/3 buffer with the following command:
`# tp 'addtobuffer SID'`
4. Import the transport to the target R/3 system with the following command:
`# tp 'import SID client=000 U16'`

The expected result of the `addtobuffer` step is a successful return code of ``0'`.

If problems occur during the `addtobuffer` step, it is likely there is a problem with the files. They may be missing, in the wrong location, or have incorrect ownership or permissions.

The expected result of the `import` step is a successful return code of either ``0'` or ``4'`. A return code of ``8'`, ``12'` or ``16'` indicates transport failure. Return codes higher than ``16'` indicate a major failure within the transport tool. If this occurs, check the present working directory to ensure the correct location. Also, check the files for existence, location, and proper ownership and access.

If problems exist during the `import`, retry the `import` step. If problems persist, check the `import` and `activation` logs for failure reason. These files are in the `/usr/sap/trans/log` location and named `.r6u` (the ``?'` stands in as a wildcard).

Importing the transports via TMS

Starting from R/3 version 4.0, SAP allows importing transports through the SAP system via transaction code `STMS`.



Security authorization in the SAP system must include proper access to import the transport request.

1. Sign-on to SAP system with a User ID that contains proper authority.
2. Execute transaction `STMS`.

3. Select the **Transport** (Truck) icon from the toolbar.
4. Select the desired system for import.
5. Add the Transport to the import queue by selecting the following path from the menu list:
Extras > Other Requests > Add
Add the transport request number to the proper import queue and execute. Re-authentication of the user's SAP User ID is likely in order to complete the step.
If an Information message is received that the "Transport request is invalid" check that the transport number was typed correctly. Otherwise, it may indicate a problem with the files. Verification of existence, location, permissions, or ownership may be needed.
6. Import the Transport request by selecting the transport number from the queue, and clicking the **Import** (Truck) icon from the toolbar. Set the target client to either `000` or any other valid client within the system and execute with the truck icon. Once again, re-authentication of the SAP User ID may be necessary. The transport will execute in asynchronous mode; a record of success or failure can be found in the transport logs.
7. The system will return to the import queue screen, where the Transport results can be checked. Select the **Logs** icon from the toolbar, or follow the menu path:
Request > Display > Logs
Locate the source system and verify all relevant logs. For this transport there should be 5 logs:
 - **DD Import**
 - **DD Activation**
 - **Import**
 - **Check Versions**
 - **ABAP/scrn. Generation**All logs should display Ended OK (return code 0) or Ended with warning (return code 4). If any logs are missing, or display a return code of 8 or higher, follow the instructions in step 6 to re-import the transport.

Upgrading/patching and uninstalling the Replicate SAP Client

The following instructions explain how to upgrade/patch, and uninstall the Replicate SAP Client.



If you are applying a patch or upgrading the Replicate for SAP Client, you should only install the main transport and the ECC-specific transport (if upgrading on an ECC system). Do not install the configuration transport again, or any customizations made to the configuration will be lost.

To upgrade or patch the Replicate SAP Client on ECC Systems

1. Apply the new main transport (InstallCode.zip).
2. Apply the new ECC-specific transport (InstallCodeECC.zip).

To upgrade or patch the Replicate SAP Client on non-ECC systems

- Apply the new main transport (InstallCode.zip).

To uninstall the Replicate SAP Client

1. Apply the "Delete Main" transport (DeleteCode.zip).
2. Apply the "Delete Package" transport (DeletePackage.zip).



Do not apply items 3 and 4 if you are also running Qlik Gold Client Solution.

3. Apply the "Delete Common" transport (DeleteCodeCommon.zip)
4. Apply the "Delete Common Package" transport (DeletePackageCommon.zip)

Managing business groups and tables

This prerequisite is only necessary if you want to edit the default Business Groups and/or tables before replicating them.

Before you can manage business groups and tables, you first need to launch the SAP Client UI.

To launch the SAP Client user interface:

1. Open your SAP client console.
2. Double-click one of the SAP Application (DB) Sources.
You will be prompted for your user name and password.
3. Enter your credentials for logging in to the selected SAP Application (DB) Source.
4. Enter /nzs4sap in the drop-down list at the top of the console and then press [Enter].
5. Click the **Business Groups Configuration** button.
A list of Business Groups is displayed in the left pane.

Managing business groups

To add a new Business Group

1. Click the **Create** toolbar button.
The **Create Business Group** dialog box opens.
2. Enter a name for your Business Group and then press [Enter].
The new Business Group is added to the **Business Groups** list in the left pane.

To duplicate a Business Group




1. Click the **Copy** toolbar button.
The **Business Group Configuration** dialog box opens.
2. In the **New Bus Object** field, enter a name for the new Business Group and then press [Enter].
The duplicated Business Group is added to the **Business Groups** list in the left pane.

To delete a Business Group




- Select the Business Group you want to delete and then click the **Delete** toolbar button.
The Business Group is deleted.

Managing tables

To add a new table to a Business Group

1. In the left pane, expand the desired Business Group.
2. Double-click the **Tables** icon.
A list of tables is shown in the right pane.
3. Click the  button above the table list to enter Edit mode.
4. Click the  button that appears to the right of the  button.
An empty row is added to the tables list.
5. Enter the **Table Name** (i.e. the virtual ABAP table) and the name of the corresponding **Source Table** (i.e. the physical table).
6. To save your changes click the **Save** button in the main toolbar.

To remove a table from a Business Group

1. In the left pane, expand the desired Business Group.
2. Double-click the **Tables** icon.
A list of tables is shown in the right pane.
3. Click the  button above the table list to enter Edit mode.
4. Select the table you want to delete.
5. Click the  button that appears to the right of the  button.
The table is deleted.
6. To save your changes click the **Save** button in the main toolbar.

Accessing tables with a non-English language

To enable replication of non-English SAP tables, add them to the configuration as follows:

1. From the zr4sap transaction, click the **Metadata Language Configuration** button.
2. Click the **Add** button and select the appropriate language.
3. Click **Save**.

To remove a previously configured language, select the desired row and click the **Delete** button.

Target collation

As SAP is case-sensitive, when a replication task is defined with SAP Application (DB) data source, the target database needs to be set up with case-sensitive collation.

Limitations and considerations

When using SAP Application (DB) as a source endpoint in a Replicate task, the following limitations apply.

- When using the HR module as a source, only transparent tables are supported.
- The STXL table is not supported.
- As the SAP HANA endpoint can be configured to connect to one instance only, in the event of failover in a HA environment, the endpoint must be manually reconfigured to point to the new instance. Then reload all associated tasks.

9 Adding and managing source endpoints

- Cluster tables are not supported when SAP uses an Oracle backend database, and the VARDATA column is defined as LONG RAW.
- Replicating data with 4-byte emoji characters is not supported.
- The DROP TABLE DDL is not supported. In order to stop capturing a captured table, you need to stop the task, remove the table from the task table list (or exclude it if it is defined as a pattern), and then resume the task. When you remove the table from the task table list, you also have the option to remove it from the corresponding Business Group.
- The CREATE TABLE DDL is not supported. When you add a new table to a Business group during a task run, and the table name corresponds to the task table list definition, the table replication will not start automatically. To ensure that the table is replicated, you need to stop the task and then resume it.
- The source_lookup Data Enrichment function is not supported. For more information on this function, see *Data Enrichment functions (page 1157)*

For information about limitations according to backend database type, see *Set up a source endpoint for SAP Application (DB) (page 499)*.

SAP Application (DB) source data types

The SAP Application (DB) endpoint for Qlik Replicate supports most SAP data types. The table below shows the SAP source data types that are supported when using Qlik Replicate and the default mapping to the Qlik Replicate data types.

For information on how the data type is mapped to the target, see the chapter for the target endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

Data types

ABAP Type	ABAP Type Description	SAP Type	Qlik Replicate Data Type
h	Table type	-	BYTES
V	Character string (old Dictionary type VARC)	-	STRING
C	Character string	-	STRING
N	Character string with only digits	-	STRING
D	Date (string: YYYYMMDD)	-	DATE
T	Time (string: HHMMSS)	-	TIME
X	Byte sequence	INT4 (4-byte integer)	I4

ABAP Type	ABAP Type Description	SAP Type	Qlik Replicate Data Type
		INT2 (2-byte integer)	I2
		INT1 (1-byte integer)	I1
		ELSE	If backend type is NUMERIC: NUMERIC If length = 0: BLOB If length > 0: BYTES
I	Integer number (4-byte integer with sign)	-	INT4
b	2-byte integer	-	INT2
s	1-byte integer	-	INT1
P	Packed number	-	NUMERIC
F	Floating point number to accuracy of 8 bytes	-	R8
g	Character string with variable length	-	STRING
y	Byte sequence with variable length	-	BLOB BYTES
u	Structured type, flat	-	BYTES
v	Structured type, deep	-	BYTES
r	Reference to class/interface	-	BYTES
i	Reference to data object	-	BYTES
n	Numeric text	NUMC	STRING

Setting general connection properties

This section describes how to set up connection parameters for your SAP Application (DB) server.

To connect to a specific SAP Application (DB) server

1. In the Qlik Replicate console, click the **Manage Endpoint Connections** toolbar button to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button.

2. In the **Name** field, enter a display name for your endpoint.
3. Optionally, in the **Description** field, enter a description for the SAP Application (DB) endpoint.
4. Select **Source** as the database **role**.
5. Select **SAP Application (DB)** as the database **Type**.



*When this endpoint is used as a duplicated source in a Log Stream Staging setup, select the **Read changes from log stream** check box and then select the relevant Log Stream Staging task from the drop-down list.
For information on setting up and managing Log Stream Staging tasks, see *Using the Log Stream* (page 1072).*

6. In the **Client** field, enter the client number of the SAP Application (DB) source you want to replicate.
7. In the **SAP schema** field, enter the name of the relevant SAP schema.
8. In the **Backend endpoint** field, click the Browse button and then select the name of the Qlik Replicate endpoint you configured earlier. See also *Set up a source endpoint for SAP Application (DB)* (page 499).

Setting advanced properties

In the **Advanced** tab, you can set internal parameters and view a summary of the endpoint settings.

Settings

Non-Unicode SAP system: Select this option if your backend database is non-unicode.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

9.30 Using SAP Extractor as a source

This section describes how to define SAP Extractor as a source endpoint in a replication task.

In this section:

- *Prerequisites (page 512)*
- *Limitations and considerations (page 530)*
- *Supported data types (page 531)*
- *Setting general connection properties (page 532)*
- *Setting advanced properties (page 535)*

Overview

The SAP Extractor endpoint leverages the SAP extractors to unload and capture SAP data directly from the SAP application, thereby eliminating the need to access the underlying database (which might not be possible due to license restrictions). Offering increased flexibility and fine-tuning options, the SAP Extractor endpoint facilitates effortless migration of SAP data to other environments.

The endpoint supports Full Load or Full Load and Apply Changes replication from standard and generic data sources.

During the replication process, Replicate does the following:

- a. Executes the extractor using RFC protocol.
- b. Stores the extracted data in an intermediate table corresponding to the source metadata structure.
- c. Accesses the intermediate table using RFC protocol to retrieve the data.
- d. Deletes the data from the intermediate table and runs the next extraction.
- e. Performs INSERT operations on the target (even for source changes that were not INSERTs).

The SAP extractor endpoint depends on the RFC protocol to copy data retrieved by ABAP programs. As a result, the SAP extractor endpoint performance is not as good as the SAP Application (DB) endpoint which accesses the underlying databases directly). Therefore, if you need to replicate huge amounts of data from SAP it is strongly recommended to use the SAP Application (DB) endpoint if possible.

Prerequisites

The following section describes the prerequisites for working with the SAP Extractor endpoint.

- *Install the SAP Java Connector (page 512)*
- *Download the required transports to the SAP Machine (page 513)*
- *Activate the extractors for Replicate (page 525)*

Install the SAP Java Connector

To work with the SAP Extractor endpoint, you need to install the SAP Java Connector on the Replicate server.

Installing the SAP Java Connector on Replicate for Windows

To install the SAP Java Connector:

1. Download the SAP Java Connector 3.x (64-bit) for Windows from the SAP Support Portal. To find out the exact SAP Java Connector version you need to support your SAP BW system, refer to the SAP BW Help.



The SAP Java Connector 3.1 running on Windows operating systems requires the Microsoft Visual Studio 2013 C/C++ runtime libraries to be installed on the system. See SAP Note 2786882 for details.

2. Install the SAP Java Connector.

3. Copy the *Sapjco3.jar* and *Sapjco3.dll* files from the SAP Java Connector installation folder to the following location under the Replicate installation directory:
<REPLICATE_INSTALL_DIR>\endpoint_srv\endpoints\SAP
4. Restart all of the Replicate services.

Installing the SAP Java Connector on Replicate for Linux

To install the SAP Java Connector:

1. Download the SAP Java Connector 3.x (64-bit) for Linux from the SAP Support Portal.
To find out the exact SAP Java Connector version you need to support your SAP BW system, refer to the SAP BW Help.
2. Unzip the file you downloaded to a folder of your choice.
The folder should contain the *Sapjco3.jar* and *libsapjco3.so* files.
3. Stop the Replicate service:

```
./areplicate stop
```
4. Copy these files to the following location under the Replicate installation directory:
<REPLICATE_INSTALL_DIR>/endpoint_srv/endpoints/SAP
5. Start the Replicate service:

```
./areplicate start
```



The SAP Extractor endpoint will not be available for selection until the Java connector libraries have been installed and the Replicate service has been restarted.

Download the required transports to the SAP Machine

This topic describes how to download and install the required transports.

Downloading the Replicate for SAP transports

Do the following:

1. Go to  [Product Downloads](#).
2. Select **Qlik Data Integration**.
3. Scroll down the **Product** list and select **Qlik Replicate**.
4. In the **Download Link** column, locate the **QlikReplicate_<version-build>_R4SAPExtractor.zip** file.
Before starting the download, check the **Version** column to make sure that the R4SAP version you are downloading corresponds to your current Replicate version.

Permissions required for installing the required transports

Replicate for SAP delivers its own authorization object: ZR4SAP. In addition to this authorization object, there are additional authorizations that need to be enabled for the Replicate software.

SAP users for Replicate for SAP Extractors

A dialog user in SAP is required to access the Replicate for SAP Extractor GUI in SAP. In addition, a communication user is required to support the RFC calls from the Replicate software to the SAP system.

Identify existing users in SAP or create dedicated users for the Replicate software.

Replicate for SAP delivers its own authorization object: /QTQVC/RAO.

9 Adding and managing source endpoints

The authorization object /QTQVC/RAO can be turned on to enable/disable specific authorizations below according to the needs of the Replicate user:

	Fixed Val.	Upper limit	Short Descript.
<input type="checkbox"/>	01		Configuration: Target Logical System
<input type="checkbox"/>	02		Configuration: Activate Extractor
<input type="checkbox"/>	03		Configuration: Manage Extractors
<input type="checkbox"/>	10		Configuration: Manage Extractors: Delete
<input type="checkbox"/>	11		Configuration: Manage Extractors: Transport
<input type="checkbox"/>	20		Delta Settings: Delete Delta Init
<input type="checkbox"/>	21		Data Extract: Schedule Extractor
<input type="checkbox"/>	22		Data Extract: Set Filter
<input type="checkbox"/>	30		Reports/Logs: View Extract logs
<input type="checkbox"/>	31		Reports/Logs: View Job Status
<input type="checkbox"/>	40		Delete Extract

Authorizations for Replicate for SAP Extractors

The SAP system needs to be open to create the logical system. This is done using SAP Transaction /nSCC4.

Make sure the **Changes and Transports for Client-Specific Objects** for the Source Client is set to **Automatic Recording of Changes** or **Changes without Automatic Recording**.

Changes and Transports for Client-Specific Objects

- Changes without automatic recording
- Automatic recording of changes
- No changes allowed
- Changes w/o automatic recording, no transports allowed

All clients need to be opened for **Cross-Client Object Changes**. Creating a Logical Client is a cross-client change that cannot be captured in a transport. It only needs to be open for a minute when the target logical system is created.

Cross-Client Object Changes

Changes to Repository and cross-client Customizing allowed

Make sure the Transaction /nSE06 **System Change Option** > **Global Setting** is set to **Modifiable** (During Logical System creation and activation of Extractors).

System Change Option

Client Setting

Global Setting: Modifiable

Administration User

SAP Authorizations for Administrator Role for initial setup of the logical system and activation of the extractors within /QTQVC/EXTREP.

Authorizations required for Administrator:

Creating the RFC.

Instead of *, RFCDEST can have the name of the logical system, which is completely configurable to the customer's preference.

Activity	Object	Value	Field
Administration for RFC Destination	S_RFC_ADM		
Administration for RFC Destination	T-HB55165600		
Activity	01, 02, 03, 06		ACTVI
Internet Communication Framework	*		ICF_VALUE
Logical Destination (Specified)	*		RFCDEST
Type of Entry in RFCDES	3, H, X		RFCTYPE

Creating the IDocs for sending to Replicate.

9 Adding and managing source endpoints

Manually	WFEDI: S_IDOCDEFT - Access to IDoc Development	S_IDOCDEFT
Manually	WFEDI: S_IDOCDEFT - Access to IDoc Development	T-HB55165600
Activity	01, 02, 03, 06	ACTVT
Extension	*	EDI_CIM
Basic type	RSSSEND, ZSQ*	EDI_DOC
Transaction Code	WE30	EDI_TCD

Creating the ZQLIK* programs and tables during activation of the extractors.

Manually	ABAP Workbench	S_DEVELOP
Manually	ABAP Workbench	T-HB55165600
Activity	01, 02, 03, 06, 07, 16	ACTVT
Package	\$TMP, Z*	DEVCLASS
Object name	ZQLIK*	OBJNAME
Object Type	DEBUG, PROG, SYST, TABL	OBJTYPE
Authorization group ABAP/4 pro	*	P_GROUP

Maintaining table information during activation and table updates.

Manually	Table Maintenance (via standard tools such as SM30)	S_TABU_DIS
Manually	Table Maintenance (via standard tools such as SM30)	T-HB55165600
Activity	02, 03	ACTVT
Table Authorization Group	*	DICBERCLS

Access to tables to check data and update as needed.

Manually	Table Access via Generic Standard Tools	S_TABU_NAM
Manually	Table Access via Generic Standard Tools	T-HB55165600
Activity	02, 03	ACTVT
Table Name	ZQLIK*	TABLE

Adding ZQLIK*-created tables to a transport.

Manually	Administration Functions in Change and Transport System	S_CTS_ADMINI
Manually	Administration Functions in Change and Transport System	T-HB55165600
Administration Tasks for Chang	TABL	CTS_ADMINI

Administering transports.

Manually	System-Specific Administration (Transport)	S_CTS_SADM
Manually	System-Specific Administration (Transport)	T-HB55165600
Administration Tasks for Chang	TABL	CTS_SADM
Logical system	*	DESTSYS
TMS: Transport Domain	*	DOMAIN

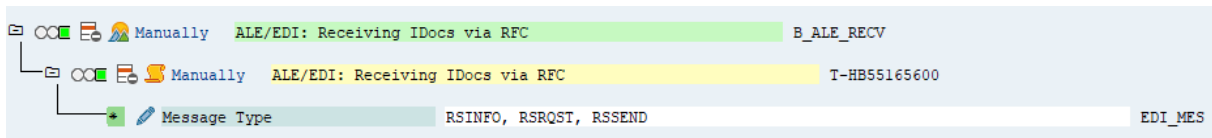
RFC Communications User

This user is needed to be used when setting up the SAP for Extractors endpoint in Replicate. It can be set as a COM user as there is no need for dialog access. Optionally you can create one dialog user ID with both the Communication user and Administration user access.

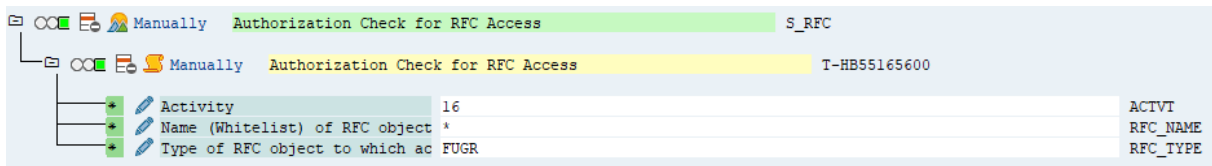
Authorizations required for RFC Communications User and Administration User:

9 Adding and managing source endpoints

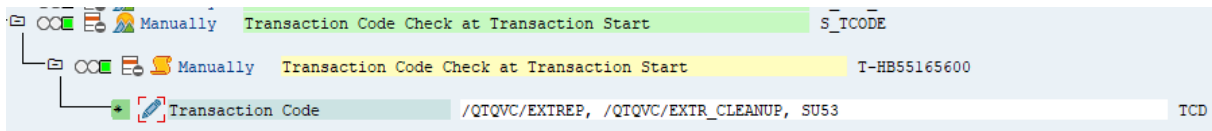
Sending and receiving of IDocs via RFC connection to Replicate.



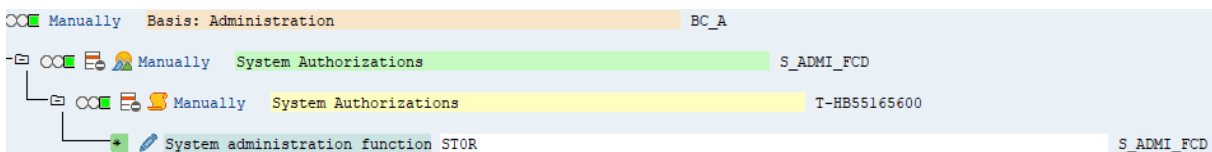
Access to the RFC communication protocol for the connection to Replicate.



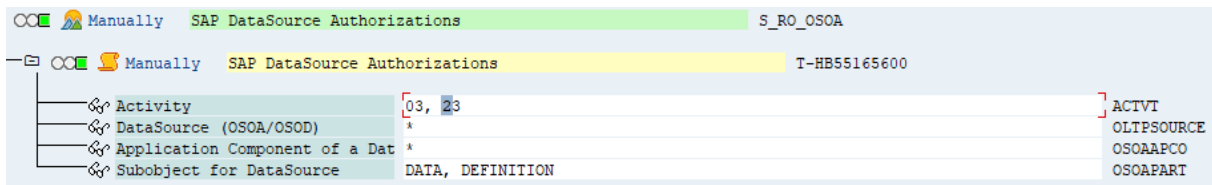
Access to the main QIQVC transactions and security checks.



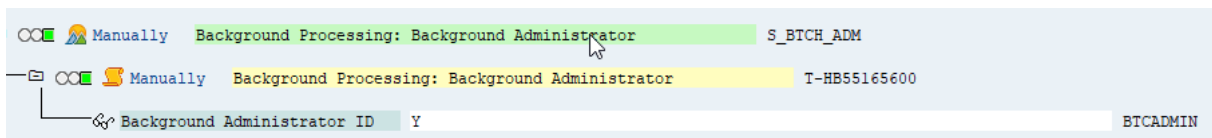
Access for trace features for the SQL functionality.



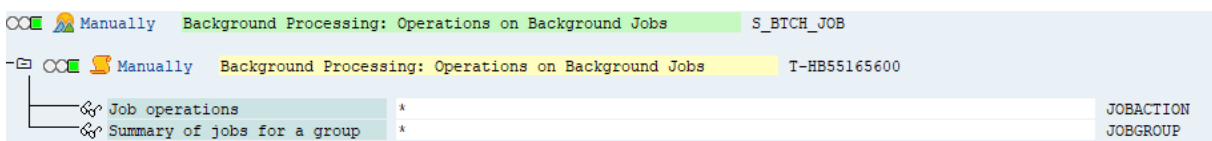
Executing the datasources during run time to extract the data from SAP.



Administrating jobs in the background as scheduled by Replicate.

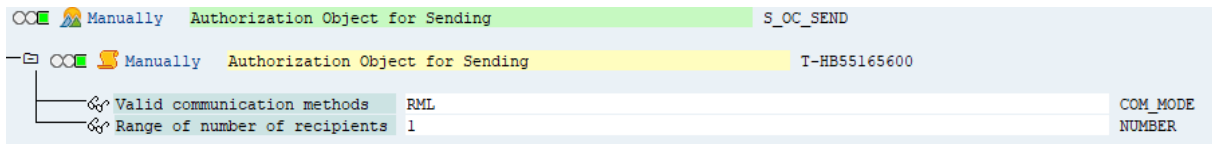


Processing jobs in the background as scheduled by Replicate.

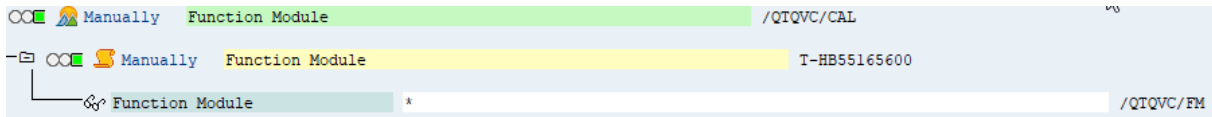


Sending internal SAP messages during the extract process.

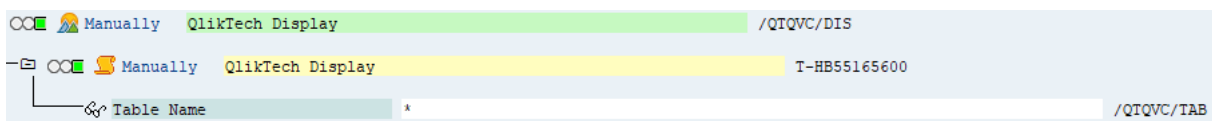
9 Adding and managing source endpoints



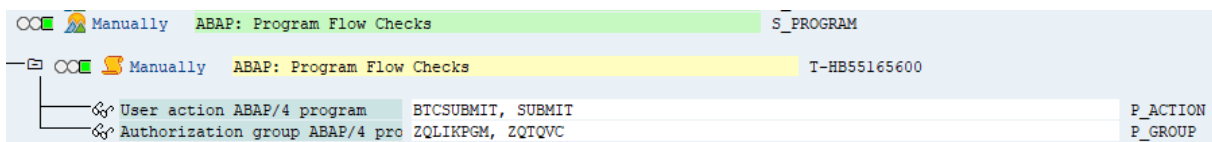
Executing functions inside the /QTQVC/* namespace to support the SQL functionality.



Viewing tables inside the /QTQVC/* namespace to support the SQL functionality.



Program authorization for Qlik programs.



Importing the required transports into the SAP system

There are two types of files required to import the ABAP objects into the SAP system: the data-file and the co-file. You also need to import the Qlik SAP connector transports.

Importing the Data-file

The data-file begins with an "R"

The data-file should be placed in the `/usr/sap/trans/datafile` system or in the directory of the server where the transports are stored.

- Typically this is a shared directory, but if not, individually place the file into that directory location for all SAP host servers.
- This file must be accessible by all systems where the Replicate for SAP Extractor is to be installed.
- Set the permissions on the file to `All` for the user, `Read` and `Execute` for the group, and `Read` and `Execute` for others.
- The owner of the file should be the `<sid>adm` user of the system to be installed. The group ownership should be `sapsys`.

Importing the Co-file

The co-file begins with a "K"

The co-file should be placed in the `/usr/sap/trans/cofiles` file system or in the directory of the server where the transports are stored.

9 Adding and managing source endpoints

- Typically this is a shared directory, but if not, individually place the file into that directory location for all SAP host servers.
- This file must be accessible by all systems where the Replicate for SAP Extractor is to be installed.
- Set the permissions on the file to `All` for the user, `Read` and `Execute` for the group, and `Read` and `Execute` for others.
- The owner of the file should be the `<sid>adm` user of the system to be installed. The group ownership should be `sapsys`.

Once the files are in the correct location, import the transport into the system using either the Operating System level transport tools (TP), or the Transport Management System (TMS) internally within SAP.

Importing the Transports via TP

1. Log on to the system at the Operating System level as the `<sid> adm`.
2. Change the directory to `/usr/sap/trans`
3. Add the transport to the R/3 buffer with the following command:

```
# tp `addtobuffer SID`
```
4. Import the transport to the target R/3 system with the following command:

```
# tp `import SID client=000 U16`
```

The expected result of the `addtobuffer` step is a successful return code of ``0``.

If problems occur during the `addtobuffer` step, it is likely there is a problem with the files. They may be missing, in the wrong location, or have incorrect ownership or permissions.

The expected result of the `import` step is a successful return code of either ``0`` or ``4``. A return code of ``8``, ``12`` or ``16`` indicates transport failure. Return codes higher than ``16`` indicate a major failure within the transport tool. If this occurs, check the present working directory to ensure the correct location. Also, check the files for existence, location, and proper ownership and access.

If problems exist during the import, retry the import step. If problems persist, check the import and activation logs for failure reason. These files are in the `/usr/sap/trans/log` location and named `.R6U` (the ``?`` stands in as a wildcard).

Importing the Transports via TMS

Starting from R/3 version 4.0, SAP allows importing transports through the SAP system via transaction code STMS.



Security authorization in the SAP system must include proper access to import the transport request.

1. Sign-on to SAP system with a User ID that contains proper authority.
2. Execute transaction `STMS`.
3. Select the **Transport** (Truck) icon from the toolbar.
4. Select the desired system for import.
5. Add the Transport to the import queue by selecting **Extras > Other Requests > Add**.

Add the transport request number to the proper import queue and execute. Re-authentication of the user's SAP User ID is likely in order to complete the step.

If an Information message is received that the "Transport request is invalid", check that the transport number was typed correctly. Otherwise, the message might indicate a problem with the files.

Verification of existence, location, permissions, or ownership may be needed.

6. Import the Transport request by selecting the transport number from the queue, and clicking the **Import** toolbar button (which has a truck icon). Set the target client to either `000` or any other valid client within the system.
 - a. Switch to the **Options** tab and make sure the following are selected:
 - Ignore Non-Permitted Transport Type
 - Ignore Non-Permitted Table Class
 - Ignore Predecessor Relations
 - Ignore Invalid Component Version
 - b. Click the **Import** toolbar button. Once again, re-authentication of the SAP User ID may be necessary. The transport will execute in asynchronous mode; a record of success or failure can be found in the transport logs.
7. The system will return to the import queue screen, where the Transport results can be checked. Select the **Logs** icon from the toolbar, or follow the menu path:

Request > Display > Logs

Locate the source system and verify all relevant logs. For this transport there should be five logs:

- **DD Import**
- **DD Activation**
- **Import**
- **Check Versions**
- **ABAP/scrn. Generation**

All logs should display `Ended OK` (return code 0) or `Ended with warning` (return code 4). If any logs are missing, or display a return code of 8 or higher, follow the instructions in step 6 to re-import the transport.

Upgrading or Uninstalling the Replicate SAP Extractor

The following instructions explain how to upgrade and uninstall the Replicate SAP Extractor.

Upgrading the Replicate SAP Extractor

To upgrade the Replicate SAP Extractor, apply the Install transport.

Uninstalling the Replicate SAP Extractor

To uninstall the Replicate SAP Extractor, apply the Delete transport.

Deleting the remaining objects

The deletion transports will remove the objects in the `/QTQVC/*` namespace.

To delete the artifacts created by the Replicate for SAP Extractors, the entries described below will need to be removed. First in the Development environment, and then transported to any other systems where they might be present (assuming they were transported).

9 Adding and managing source endpoints

Assignment of source systems to BIW systems including IDoc type (your SLOGSYS will be different) –
RSBASIDOC

This entry will be removed using function RSAP_BIW_DISCONNECT with the following parameters (using our example):

Table: RSBASIDOC
Displayed Fields: 12 of 12 Fixed Columns: [2] List Width 0250

SLOGSYS	RLOGSYS	OBJSTAT	BIDOCYP	TSIDOC3X	TSPREFIX	SRCTYPE	SAPREL	TSTPNM	TIMESTMP	SBWBCRL	RBWBCRL
T90CLNT090	ZGOLDTEST1	ACT	ZSQAZGOLDTEST1	000	QA	3	750	VHAMMOND	20,210,512,191,023	750	30B

Function module **RSAP_BIW_DISCONNECT**
 Uppercase/Lowercase

RFC target sys:

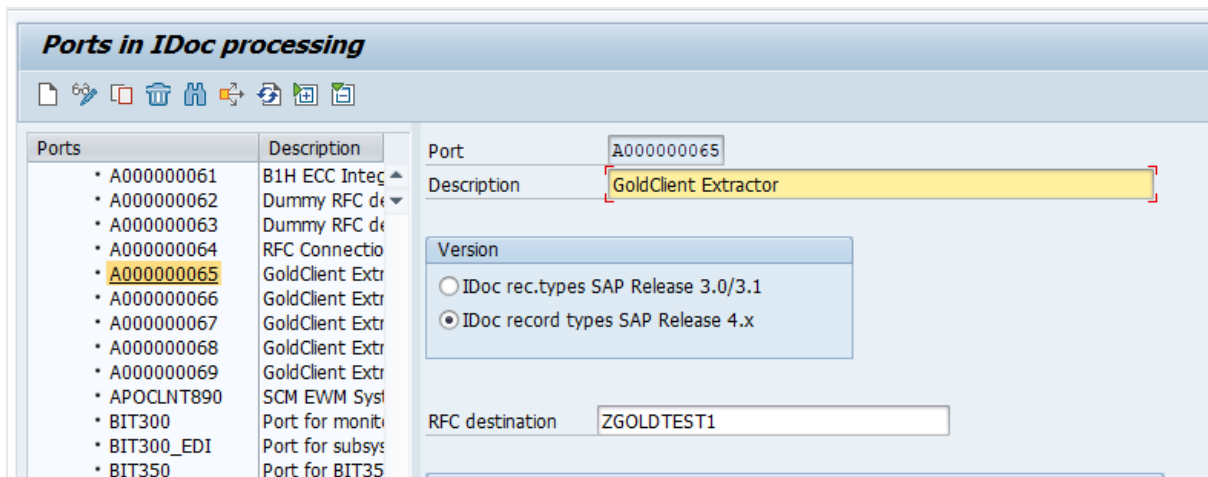
Import parameters	Value
I_BIW_LOGSYS	ZGOLDTEST1
I_OLTP_LOGSYS	T90CLNT090
I_FORCE_DELETE	<input checked="" type="checkbox"/>

Transaction WE20

Select the Logical System created for the Replicate for SAP Extractors, and then delete.

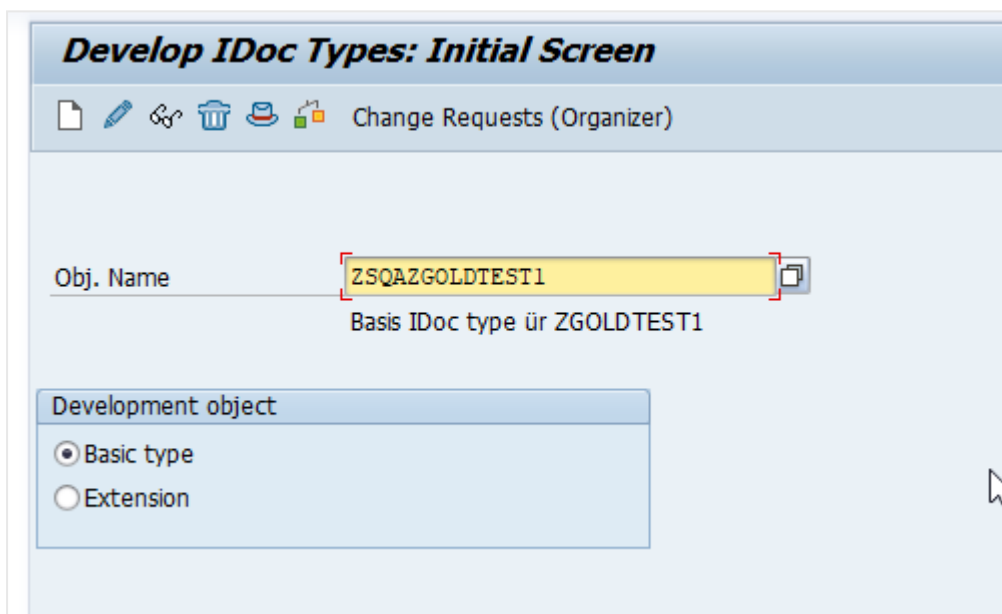
Transaction WE21

Select the IDoc Port used in the partner profile in WE20, and then delete.



Transaction WE30

Select the IDoc Port used for message type RSEND in the partner profile in WE20, and then delete.



Below are the logical system entries that need to be deleted and collected in a transport.

Table TBDLS

Table: TBDLS
 Displayed Fields: 6 of 6 Fixed Columns: 1 List Width 0250

LOGSYS	ISPRFCDES	ISPRFCDESE	ISPREL	ISPRFCDESE	STEXT
ZGOLDTEST1					GoldClient Extractor

Table TBDLST

Table: TBDLS
 Displayed Fields: 6 of 6 Fixed Columns: [1] List Width 0250

LOGSYS	ISPRFCDES	ISPRFCTYPE	ISPREL	ISPRFCDESF	STEXT
ZGOLDTEST1					GoldClient Extractor

RFC Destination entries as well as ports and IDOC entries are listed below.

You can use R3TR TABU to manually delete the following table entries into a Customizing request

Table RFCDES

Table: RFCDES
 Displayed Fields: 4 of 20 Fixed Columns: [1] List Width 0250

RFCDDEST	RFCTYPE	RFCHOPTIONS
ZGOLDTEST1	3	Y=2,h=2,z=-2,W=Y,B=N,C=N,E=N,T=N,K=Y,s=N,u=N,q=0,i=N,b=N,k=N,R=N,j=N,

Table RFCDOC

Table: RFCDOC
 Displayed Fields: 4 of 5 Fixed Columns: [2] List Width 0250

RFCLANG	RFCDDEST	RFCDDOC1
E	ZGOLDTEST1	GoldClient Data Extractor Interface *please do not modify*

Table RFCATTRIB

Table: RFCATTRIB
 Displayed Fields: 10 of 10 Fixed Columns: [1] List Width 0250

RFCDDEST	CUNAME	CUDATE	MUNAME	MUDATE	RFCDVERSION	CUCLIENT	MUCLIENT	CUTIME	MUTIME
ZGOLDTEST1	VHAMMOND	12.11.2020	VHAMMOND	12.11.2020		800	800	20:05:11	20:05:11

The following entries should be deleted with transaction WE20, but you should confirm they are removed and collected in a transport.

Partner Profile Table – EDPP1

Data Browser: Table EDPP1 Select Entries 1

Check Table...

Table: EDPP1
 Displayed Fields: 10 of 10 Fixed Columns:

MANDT	PARNUM	PARTYP	CLASS	MATLVL	TEST	ARCHIV	USRTYP
800	ZGOLDTEST1	LS	SE	A			US

Outbound Parameters – EDP13

Data Browser: Table EDP13 Select Entries 2

Table: EDP13
 Displayed Fields: 18 of 30 Fixed Columns: 8

	MANDT	RCVPRN	RCVPRT	RCVPFC	MESTYP	MESCOD
<input type="checkbox"/>	800	ZGOLDTEST1	LS		RSINFO	
<input type="checkbox"/>	800	ZGOLDTEST1	LS		RSSEND	

Inbound parameters – EDP21

Data Browser: Table EDP21 Select Entries 1

Table: EDP21
 Displayed Fields: 16 of 16 Fixed Columns: 8

	MANDT	SNDPRN	SNDPRT	SNDPFC	MESTYP	MESCOD
<input type="checkbox"/>	800	ZGOLDTEST1	LS		RSRQST	

The following entries should be deleted with transaction WE21, but you should confirm they are removed and collected in a transport.

Port Header Table - EDIPOA

Table: EDIPOA
 Displayed Fields: 7 of 7 Fixed Columns: 2 List Width 0250

	MANDT	PORT	LOGDES	VERSIO	USE_QUEUE	SEND_SMALL	TARGET_DOCREL
<input type="checkbox"/>	800	A000000065	ZGOLDTEST1	3			

Port Table – EDIPOINT

Data Browser: Table EDIPOINT Select Entries 1

Table: EDIPOINT
 Displayed Fields: 4 of 4 Fixed Columns: 2

MANDT	PORT	PORTTYP	DESCRI
800	A000000065	1	GoldClient Extractor

The following entries should be deleted with transaction WE30, but you should confirm they are removed and collected in a transport.

IDOC Message type – EDIMSG

Data Browser: Table EDIMSG Select Entries 1

Table: EDIMSG
 Displayed Fields: 5 of 5 Fixed Columns: 3 List Width 0250

MESTYP	IDOCTYP	CIMTYP	RELEASED	ACTFLAG
RSSSEND	ZSQAZGOLDTEST1			

Deleting the ZQLIK* Tables

Select and then delete all the entries in “Manage extractors”

Select	DataSource	Target Logsys	Table name	Data Extract Program Name	Data Transfer Program Name	Init
<input checked="" type="checkbox"/>	0ACCOUNT_ATTR	ZGOLDTEST1	ZQLIKDTCA000011	ZQDS003YQ28BEP9MHV80Y6OZ02HB1	ZQDT003YQ28BEP9MHV80Y6OZ02HB1	
<input checked="" type="checkbox"/>	0ACCOUNT_TEXT	ZGOLDTEST1	ZQLIKDTCA000012	ZQDS003YQ28BEP9MHV80Y6OZ05ZQ5	ZQDT003YQ28BEP9MHV80Y6OZ05ZQ5	
<input checked="" type="checkbox"/>	0AC_DOC_TYP_TEXT	ZGOLDTEST1	ZQLIKDTCA000013	ZQDS003YQ28BEP9MHV80Y6OZ05ZQ5	ZQDT003YQ28BEP9MHV80Y6OZ05ZQ5	
<input checked="" type="checkbox"/>	0ARTICLE_ATTR	ZGOLDTEST1	ZQLIKDTCA000014	ZQDS003YQ28BEP9MHV80YHT8U6TCT	ZQDT003YQ28BEP9MHV80YHT8U6TCT	
<input checked="" type="checkbox"/>	0ART_ST_LOC_ATTR	ZGOLDTEST1	ZQLIKDTCA000015	ZQDS003YQ28BEP9MHV80YHT8UA5GD	ZQDT003YQ28BEP9MHV80YHT8UA5GD	
<input checked="" type="checkbox"/>	0ASSET_AFAB_ATTR	ZGOLDTEST1	ZQLIKDTCA000016	ZQDS003YQ28BEP9MHV80YB9MW5YZI	ZQDT003YQ28BEP9MHV80YB9MW5YZI	
<input checked="" type="checkbox"/>	0ASSET_AFAB_TEXT	ZGOLDTEST1	ZQLIKDTCA000017	ZQDS003YQ28BEP9MHV80YB9MW5YZI	ZQDT003YQ28BEP9MHV80YB9MW5YZI	
<input checked="" type="checkbox"/>	0ASSET_ATTR	ZGOLDTEST1	ZQLIKDTCA000018	ZQDS003YQ28BEP9MHV80YB9MW5YZI	ZQDT003YQ28BEP9MHV80YB9MW5YZI	
<input checked="" type="checkbox"/>	0ASSET_ATTR_TEXT	ZGOLDTEST1	ZQLIKDTCA000019	ZQDS003YQ28BEP9MHV80YVGA3A5	ZQDT003YQ28BEP9MHV80YVGA3A5	
<input checked="" type="checkbox"/>	0ASSET_CLAS_TEXT	ZGOLDTEST1	ZQLIKDTCA000020	ZQDS003YQ28BEP9MHV80YVGA3A5	ZQDT003YQ28BEP9MHV80YVGA3A5	
<input checked="" type="checkbox"/>	0ASSET_MAIN_TEXT	ZGOLDTEST1	ZQLIKDTCA000021	ZQDS003YQ28BEP9MHV80YVGA3A5	ZQDT003YQ28BEP9MHV80YVGA3A5	

Collect all of the deleted ZQLIK* TABU entries and the ZQD* PROG program entries if you need to transport them.

Activate the extractors for Replicate

After importing the transports, you need to activate them before configuring the Replicate SAP Extractor endpoint.





As the transport installations are not automatically validated by Replicate, it is strongly recommended to validate the installations before running tasks with the SAP Extractor endpoint.

To do this:




9 Adding and managing source endpoints

1. Open the Replicate for SAP Extractors launch pad through transaction /QTQVC/EXTREP.


Replicate Extractor Endpoint







Configuration

-  Target Logical System
-  Activate Extractors
-  Manage Extractors



Delta Settings

-  Delete Delta Init


Data Extract

-  Create Extract Filter
-  Adjust Extract Filter
-  Schedule Extract Run
-  Delete Extract Data

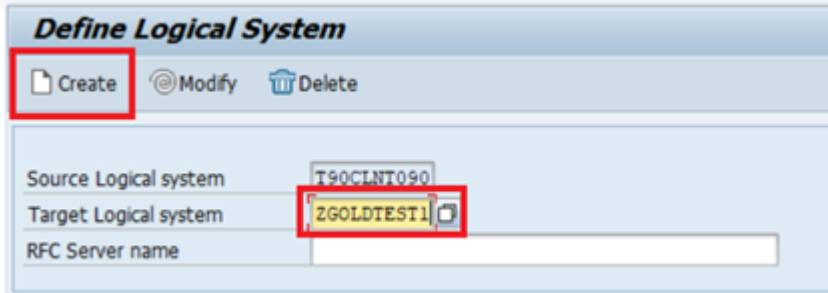
Reports/Logs

-  View Extract Logs
-  View Job Status

2. Create a logical system for the Extractor solution to use as follows:

 This should be separate from any other Logical System present in the system.

- a. In the **Configuration** section, click the **Target Logical System** button.
- b. In the **Target Logical system** field, enter a technical name for the Logical System and then click **Create**.



Define Logical System

Source Logical system: T90CLNT090


Target Logical system: ZGOLDTEST1

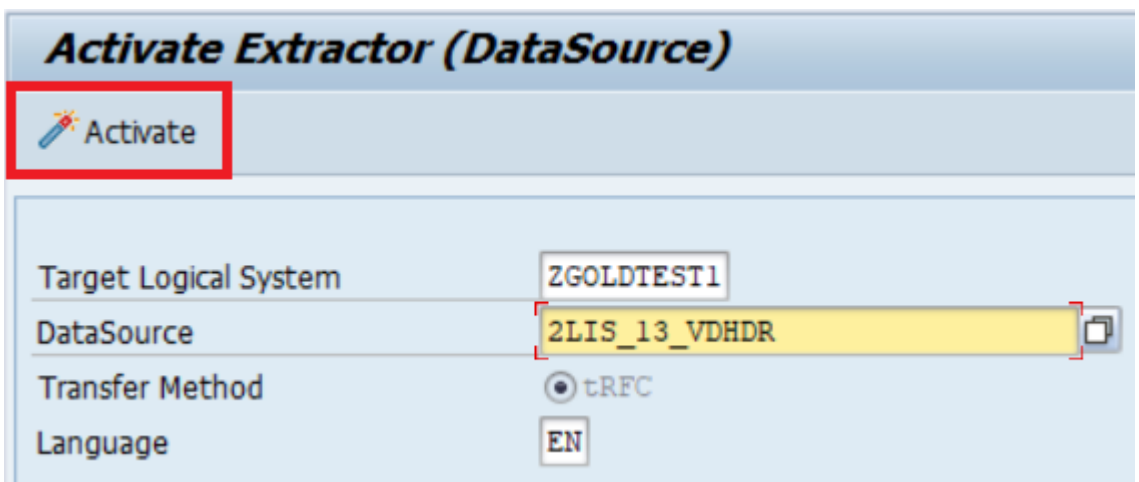
RFC Server name: _____

This will create the Logical System and setup its associated RFC connections for use by the Replicate endpoint.

The following confirmation should be shown:
Successfully created logical system.

3. Activate the extractors to associate them with the Logical System as follows:
 - a. In the **Configuration** section, click the **Activate Extractors** button.
 - b. Using the Logical System created earlier, select the extractor (also known as **DataSource**) you wish to activate from the list of available extractors.

-  • An extractor may be missing from the list if it has not yet been activated in the ECC system in transaction RSA5.
- Activated extractors will be recorded in the ROOSPRMSC and ROOSPRMSF tables associated with the Logical System.



Activate Extractor (DataSource)

Target Logical System: ZGOLDTEST1

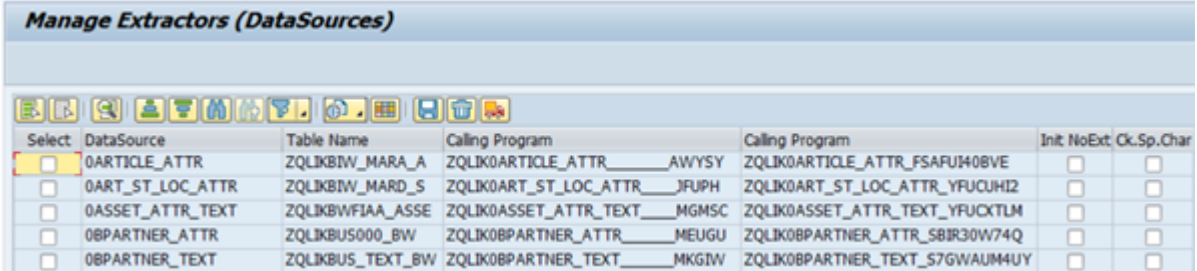
DataSource: 2LIS_13_VDHDR

Transfer Method: tRFC

Language: EN

- c. Click **Activate** to start the activation process.
You should see a confirmation message at the bottom of your screen.

- d. Repeat steps b-c for any additional extractors you wish to activate.
- e. To confirm successful activation of the extractors, click **Manage Extractors**.
The **Manage Extractors** window opens.



Select	DataSource	Table Name	Calling Program	Calling Program	Init	NoExt	Ck.Sp.	Char
<input checked="" type="checkbox"/>	0ARTICLE_ATTR	ZQLKBIW_MARA_A	ZQLK0ARTICLE_ATTR_____AWYSY	ZQLK0ARTICLE_ATTR_FSAFUH08VE	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	0ART_ST_LOC_ATTR	ZQLKBIW_MARD_S	ZQLK0ART_ST_LOC_ATTR_____JFUPH	ZQLK0ART_ST_LOC_ATTR_YFUCUH12	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	0ASSET_ATTR_TEXT	ZQLKBWFAA_ASSE	ZQLK0ASSET_ATTR_TEXT_____MGMSC	ZQLK0ASSET_ATTR_TEXT_YFUCXTLM	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	0BPARTNER_ATTR	ZQLKBUS000_BW	ZQLK0BPARTNER_ATTR_____MEUGU	ZQLK0BPARTNER_ATTR_SBR30W74Q	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	0BPARTNER_TEXT	ZQLKBUS_TEXT_BW	ZQLK0BPARTNER_TEXT_____MKGIV	ZQLK0BPARTNER_TEXT_S7GWAUM4UY	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

In the **Manage Extractors** window, you can verify which extractors have been activated for your Logical System and manage their use.

Once you have confirmed that the required extractors have been activated, you can then configure the SAP Extractor endpoint settings in Replicate.

Target collation

As SAP is case-sensitive, when a Replicate task is defined with a SAP Extractor source, the target endpoints need to be set up with case-sensitive collation.

Limitations and considerations

When using SAP Extractor as a source endpoint in a Replicate task, the following limitations apply.

- DDLs are not supported. Should you need to change the extractor:
 - a. Stop the task and remove the extractor from it.
 - b. Replace the extractor and activate it.
 - c. Add the extractor to the task again and resume.
- Each extractor can be associated with a single task only. Additionally, the extractor should be exclusively run by Replicate.
- Table transformations and filters may impact performance as they are not propagated to the extractors.
- Source lookup is not supported.
- Replicate supports only one instance of each extractor. Consequently, each extractor should be associated with only one Logical System.
- SAP Extractor uses the metadata extracted from the extractor's Z-table. Consequently, the target tables will be created with this table's columns and without a Primary Key/Unique Index.
- All changes are captured as INSERT operations.
- The [Log Stream feature](#) is not supported with this endpoint.
- Replicating data with 4-byte emoji characters is not supported.
- The **Apply Changes** and **Store Changes** replication options cannot be enabled without the **Full Load** replication option.
- The "Start from timestamp" and "Start from position" options for running tasks are not supported.
- Replicate header fields are not supported.

- The [Allow unlimited LOB size and Optimize handling when LOB size is less than](#) LOB handling options are not supported.

Supported data types

The table below shows the SAP Extractor source data types that are supported when using Qlik Replicate and the default mapping to the Qlik Replicate data types.

For information on how the data type is mapped to the target, see the chapter for the target endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).

Data types

ABAP Data Type	Qlik Replicate Data Type
ACCP	DATE
CHAR	STRING
CLNT	STRING
CUKY	STRING
CURR	NUMERIC(15.2)
DATS	DATE
DEC	REAL8
FLTP	REAL4
INT1	INT1
INT2	INT2
INT4	INT4
LANG	STRING
NUMC	STRING
QUAN	REAL8
RAW	BYTES
SSTR	STRING
STRG	CLOB
TIMS	TIME
UNIT	STRING
VARC	STRING

Setting general connection properties

This section describes how to set up connection parameters for a specific SAP Extractor server or for a SAP Extractor system using load balancing.

Connecting to a specific SAP Application server

To connect to a specific SAP Application server:

1. In the Qlik Replicate console, click the **Manage Endpoint Connections** toolbar button to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button.
2. In the **Name** field, enter a display name for your endpoint.
3. Optionally, in the **Description** field, enter a description for the SAP Application endpoint.
4. Select **Source** as the database **role**.
5. Select **SAP Extractor** as the database **Type**.
6. From the **Connection mode** drop-down list, select **Custom Application Server**.
7. In the **Server name** field, enter the IP address of the Application Server on which the SAP Extractor source is located.
8. In the **Instance identifier** field, enter the instance number of the SAP Extractor source you want to replicate.
9. In the **Client identifier** field, enter the System ID of the SAP Extractor source you want to replicate.
10. Enter your credentials (**User Name**, **Password**) for accessing the SAP Extractor source. These are the credentials for the communication user created earlier in SAP.
11. From the **Number format** drop-down list, select one of the following, according to the number format in your system (Replicate needs this information in order to be able to parse the STRING correctly):
 - 1,234,567.89
 - 1.234.567,89
 - 1 234 567.89

Connecting to a SAP system using load balancing

To connect to a SAP system using load balancing:

1. In the Qlik Replicate console, click the **Manage Endpoint Connections** toolbar button to open the **Manage Endpoints Connections** dialog box.
2. In the **Name** field, enter a display name for your endpoint.
3. Optionally, in the **Description** field, enter a description for the SAP Application endpoint.
4. Select **Source** as the database **role**.
5. Select **SAP Extractor** as the database **Type**.
6. From the **Connection mode** drop-down list, select **Load Balancing Server**.
7. In the **Message server** field, enter the host name or IP address of the message server host.
8. In the **Application servers group name** field, enter the name of the SAP server group. This is an optional group of application servers in a load balancing connection.
9. In the **SAP system name** field, enter the SAP R/3 name.

10. In the **Message server service** field, enter the name of the SAP message server service as specified in the following file:
`<system drive>:\WINDOWS\system32\drivers\etc\services`
If you do not specify a value, the Data Provider for SAP uses the following default name:
`sapms<R/3 system name>`
11. In the **Client identifier** field, enter the System ID of the SAP Application source you want to replicate.
12. Enter your credentials (**Username, Password**) for accessing the SAP Extractor source.
Note: These are the credentials for the communication user created earlier in SAP.
13. From the **Number format** drop-down list, select one of the following according to the number format in your system (Replicate needs this information in order to be able to parse the STRING correctly):
 - 1,234,567.89
 - 1.234.567,89

Security

In the **Security** settings, you can configure Secure Network Communication (SNC).

Prerequisites for working with SNC

Follow the steps below to install the Secure Network Communication (SNC) client on the Replicate machine.

What you need:

- An exported certificate (.crt) of the SAP server
- SAPCAR.EXE
- SAP user (authorized customer)
- The version of the crypto library which is installed on the corresponding SAP server

Installing the SNC client

1. Create a workspace folder for the SAP SNC files and binaries (hereafter referred to as "your SNC folder"), for example: "C:\snc\"
2. Copy the exported server certificate and **SAPCAR.EXE** to your SNC folder.
3. Go to <https://support.sap.com/en/my-support/software-downloads.html> and search for **SAPCRYPTOLIB** under **Installations & Upgrades**. Download the 64-bit **.SAR** to your SNC folder.
4. Open a command prompt and change the working directory to your SNC folder. Then run the following command to unpack the content of the **.SAR** to your SNC folder:

```
sapcar -xvf LibName.sar
```

Example:

```
sapcar -xvf SAPCRYPTOLIBP_8541-20011731_32.SAR
```

5. Add system environment variables as follows:
 - a. Add a system environment called **SECUDIR** with the path to your SNC folder as its value.
 - b. Add a system environment variable called **QLIK_SNC_LIB** with the path to the **sapcrypto.dll** file as its value.
 - c. Add the newly added environment variables to the "PATH" environment variable.
6. Determine the **<PSE_File_Name>** and choose a **<PSE_PIN>** to protect it. You will need to provide this information in the next steps.

Example:

```
pseName: "CN=USR,OU=SAP,O=Qlik,C=IS" password: password123
```

7. Determine the **<SNC_NAME>**. It should look something like this: CN=USR, OU=SAP, O=Qlik, C=IS
See also *Determining the server SNC name (page 534)* below.
8. Make sure you have the required permissions to access and execute the files in the **SECUDIR** folder, and then run the following command to generate the PSE file:

```
sapgenpse get_pse -p <PSE_File_Name>.pse -x <PSE_PIN> <SNC_NAME>
```

Example:

```
sapgenpse get_pse -p usr.pse -x password123 "CN=USR,OU=SAP,O=Qlik,C=IS"
```
9. Bind the PSE file with the OS user and create the **CRED_V2** file in **SECUDIR** folder as follows:
 - a. Make a note of the OS user under which Replicate is running:
 - Windows: Open the Services console and double-click the Qlik Replicate Server service to open the properties dialog. Look in the **Log On** tab.
 - Linux: Run the `ps aux` command.
 - b. Run the following command:

```
sapgenpse seclogin -p <PSE_File_Name>.pse -x <PSE_PIN> -O <OS_USER>
```

Example:

```
sapgenpse seclogin -p usr.pse -x password123 -O SYSTEM
```
10. Generate the CRT file by executing the following command:

```
sapgenpse export_own_cert -o <PSE_File_Name>.crt -p <PSE_File_Name>.pse -x <PSE_PIN>
```

Example:

```
sapgenpse export_own_cert -o usr.crt -p usr.pse -x password123
```
11. Import the SAP Application Server Certificate (**<SERVER_CERT>**) to the PSE by executing the following command:

```
sapgenpse maintain_pk -a <SERVER_CERT>.crt -p <PSE_File_Name>.pse -x <PSE_PIN>
```

Example:



```
sapgenpse maintain_pk -a sapsys.crt -p usr.pse -x password123
```
12. To verify that the DN of the SAP Server's PSE was imported into the client, run the following command and then check the "subject" value:

```
sapgenpse maintain_pk -v -l -p <PSE_File_Name>.pse
```

Example:

```
sapgenpse maintain_pk -v -l -p usr.pse
```

Importing the client certificate

1. Connect to the SAP Application Server and navigate to the "STRUST" transaction using an authorized user.
2. Double-click the **SNC (SAPCryptolib)** folder.
3. Click  to switch to Change view.
4. Click  to import the certificate.
5. In the new dialog, enter the path to the .crt file that was created earlier, then click continue.
6. Verify the details of the certificate in the **Certificate** section.
7. Click **Add to Certificate List** to add the certificate to the list.
8. Save the changes.

Determining the server SNC name

There are two ways you can determine the server name:

- **Method 1:** Decrypt the server CRT file using the OpenSSL command. The server name will be part of the subject.
- **Method 2:** This method requires appropriate permissions. While connected to the system:
 1. Run the RZ10 transaction.
 2. Select the system profile.
 3. Select the **Extended Maintenance** option and then click **Display**.
 4. The value of the `snc/identity/as` parameter should be the SNC name.

Connection settings

Configure the SNC settings in the SAP Extractor endpoint as follows:

- **Activate Secure Network Communication:** Select to turn on SNC.
- **SNC name:** The SNC partner name.
Example:
`p:CN=SYS, OU=SAP, O=Qlik, C=IS`
- **SNC quality of protection** - Select one of the following:
 - **Authentication only:** Select to verify the identity of the SAP Extractor machine. This is the minimum protection level offered by SNC.
 - **Integrity protection:** Select to detect any changes or manipulation of the data, which might have occurred between the Replicate machine and the SAP Extractor machine.
 - **Privacy protection:** Select to encrypt the messages being transferred to prevent eavesdropping. Privacy protection also includes integrity protection. This is the maximum level of protection provided by SNC.
 - **Maximum security available:** The maximum level of data protection supported by the SAP Extractor machine.

Setting advanced properties

In the **Advanced** tab, you can set the following parameters:

- **Extractor mode** - Choose one of the following:
 - **Continuous extraction** - Delta is extracted according to the **Run extractor every (minutes)** interval.
 - **Run extractor every (minutes)** - Specify how often you want Replicate to check for changes.
 - **Intermittent extraction** - This option is useful if the current delta needs to be processed or consumed *before* any subsequent delta is extracted. In this mode, all extractors will be run once and a file indicating the completion status of the extraction process will be written to the **Extract Completed Directory** folder.
To run additional extractions, stop the task and then resume it.
When Full Load completes, the task creates a file for each table with the following name: **<table-name>.txt**. The file indicates the Full Load completion status.
After each delta extraction, the task creates a file for each table with the following name: **cdc_<task-name>_<timestamp>.txt**. The file name reflects the task name and the extraction completion timestamp.



The first time the task is run, only the Full Load will be extracted. Once the Full Load completes, stop and resume the task to extract the delta.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

9.31 Using SAP HANA as a source

This section describes how to set up and use a SAP HANA endpoint as a source in a replication task.



The SAP HANA endpoint can either be accessed directly or via the SAP Application endpoint. For an explanation of how to set up the SAP Application endpoint, see [Using SAP Application as a source \(page 483\)](#)

In this section:

- [Prerequisites \(page 538\)](#)
- [Limitations and considerations \(page 540\)](#)
- [Required permissions \(page 541\)](#)
- [Supported data types \(page 542\)](#)
- [Setting general connection properties \(page 543\)](#)
- [Setting advanced properties \(page 544\)](#)
- [Removing Replicate artifacts from the database \(page 548\)](#)

Prerequisites

Before SAP HANA can be used as a source endpoint in a replication task, the prerequisites described in this section must be met.



SAP HANA ODBC driver 2.13 and later are not supported.

Port

Open inbound port number 3xx15 to the SAP HANA server where xx is the instance number of the SAP HANA database that contains the source tables.

For example, the port for instance 90 would be 39015.

Required clients

Windows

Install the SAP HANA ODBC 64-bit Driver 2.x for Windows on the Replicate Server machine. The driver name is HDBODBC.

Linux

Download the SAP HANA ODBC 64-bit Driver 2.x from the [SAP Software Download Center](#). You must have an account with SAP to download the software.

Example driver name: HDBODBC

Then, open a shell prompt and do the following:

1. Stop the Replicate service:
`./areplicate stop`
2. Optionally, confirm that the service has stopped:
`./areplicate status`
3. Install the SAP HANA ODBC 64-bit Driver 2.x for Linux on the Replicate machine.
4. Add the following section to the `odbcinst.ini` file located in directory `/etc`:
[HDBODBC]
Description=64-bit HANA ODBC Driver
Driver=/opt/sap/hdbclient/libodbcHDB.so
fileusage=1
5. Start the Replicate service:
`./areplicate stop`
6. Optionally, confirm that the service has started:
`./areplicate status`

Capturing data changes

Changes can be captured from SAP HANA using either of the following methods:

- Trigger-based
- Log-based

The prerequisites for both of these methods are described below.

Trigger-based

Replicate requires the following artifacts to be created in the source database (either manually by the DBA or automatically by Replicate):

- **Three triggers for each of the designated source tables:** The triggers capture changes to the source tables (INSERTs, UPDATEs, and DELETEs) and write them to the Replicate `attrep_cdc_changes` table described below. A separate trigger is required for each DML operation. The triggers will be created in the source table schema.
- **The Replicate `attrep_cdc_changes` table:** This table will contain changes captured by the triggers and will either be created in the schema of the user specified in the **General** tab of the SAP HANA endpoint settings, or in the [schema specified in the endpoint settings' Advanced tab](#).



As **Replicate** periodically scans the `attrep_cdc_changes` table for changes, the number of UPDATEs shown in the monitor's **Change Processing** tab reflects the number of UPDATEs found (in the `attrep_cdc_changes` table) at the time of scanning. This may not be the actual number of UPDATEs performed on the source since UPDATEs are sometimes merged with INSERTs (depending on when the `attrep_cdc_changes` table was last scanned).

- **The Replicate `attrep_cdc_log` table:** When the **Use log table** option in the SAP HANA endpoint's **Advanced** tab is enabled, changes are copied from the `attrep_cdc_changes` table to the `attrep_cdc_log` table. During the task, Replicate reads the changes from the `attrep_cdc_log` table and deletes the old data according to the cleanup settings (specified in the endpoint's **Advanced** tab). The table will either be created in the schema of the user specified in the **General** tab of the SAP HANA endpoint settings, or in the [schema specified in the endpoint settings' Advanced tab](#).

To create the CDC artifacts automatically (the default):

- When defining the SAP HANA source endpoint, make sure that the **Create CDC Artifacts** check box in the **Advanced** tab is selected (the default).

To create the CDC artifacts manually:

1. Define the task as you would any other Replicate task, but when configuring the SAP HANA source endpoint, clear the **Create CDC Artifacts** check box in the **Advanced** tab.
2. Once the task is defined, select any of the **Metadata only** options in the **Advanced Run Options** window, and then click **OK**.
The task will run and generate the scripts for creating the CDC artifacts in the following location and then stop:
`~/data/tasks/<task_name>/scripts`
3. Execute the scripts in the source database.
4. Run the Replicate task normally.



Capturing changes from SAP HANA requires the source tables to have a Primary Key or Unique Index.

Log-based

- Supported from SAP HANA 2.0 SPS 04
- Redo and backup logs:
 - The logs must be in their original format only
 - The log folders must be shared with the Replicate machine.

See also: *Required permissions (page 541)*

Limitations and considerations

The following limitations apply when using SAP HANA as a source.

General limitations

- Change Processing is not supported with SAP BW tables.
- The **AR_H_USER** header column is not supported.
- Compatibility views are not supported and will not be available for selection.
- CA views are not supported and will not be available for selection.
- Row store data storage is not supported.
- Multitenant architecture is only supported on a single host.
- Only transparent tables (including SAP S/4HANA transparent tables) are supported.
- Latency when applying changes - The latency shown in the User Interface is the interval between Replicate capturing a change from the source (i.e. not the source commit time), and that change existing on the target.

Limitations and considerations when using the Trigger-based CDC method

- Changes to a table's Primary Key will not be replicated to the target.
- By default, the longest supported transaction length is 60 seconds. If there are transactions in your environment that exceed 60 seconds, set the **Transactional consistency** value in the endpoint's **Advanced** tab to the required value.
- For tasks configured with the Apply Changes replication option, transformation filters only support Primary Key columns.
- Latency when applying changes - The latency shown in the User Interface is the interval between Replicate capturing a change from the source (i.e. not the source commit time), and that change existing on the target.
- DDL operations are not supported.
- Soft DELETE operations are not supported and will not be captured.
- The [Apply Conflicts error handling policy](#) must be set to the following:
No record found for applying an UPDATE: INSERT the missing target record
- The [Store Changes](#) replication mode is not supported when working with Change Tables. It is only supported when working with an Audit table.

Limitations and considerations when using the Log-based CDC method

- Replicate creates a `hana__row_id` column on the target as a Unique Index and changes the original Primary Key/Unique Index to a standard column.
- Capturing DELETE and UPDATE operations for [filtered records](#) is not supported.
- Transformations that affect multiple columns are not supported.
- When storing changes in Change Tables, only the Row_id and changed columns of the after-image will be stored. All other columns will have NULL values.
- The TRUNCATE table and PARTITION DDLs are not supported. On detecting a table with such operations, Replicate will suspend the table and require it to be reloaded.
- As the SAP HANA endpoint can be configured to connect to one instance only, in the event of failover in a HA environment, the endpoint must be manually reconfigured to point to the new instance. Then reload all associated tasks.
- The following [Apply Conflicts error handling policy](#) is not supported:
No record found for applying an UPDATE: INSERT the missing target record
- The following [Change Processing Tuning](#) option is not supported:
Using the **Apply changes using SQL MERGE** option in **Batch optimized apply** mode

Unsupported data types

The following data types are not supported:

- ARRAY
- ST_GEOMETRY
- ST_POINT

Required permissions

This section lists which permissions are required for Replicate to be able to capture changes from SAP HANA.

When using trigger-based CDC

- Grant SELECT and TRIGGER on the SAP schema containing the selected source tables to the user specified in the SAP HANA source endpoint.
- If the `attrep_cdc_changes` table is created in a schema owned by the user specified in the SAP HANA source endpoint settings (the default), no further permissions are required.
- If the `attrep_cdc_changes` table is created in a schema not owned by the user specified in the endpoint settings, grant SELECT, INSERT, UPDATE and DELETE on the `attrep_cdc_changes` table to the specified user.

See also: *Capturing data changes (page 538)*

When using log-based CDC

- Grant SELECT permission on the selected source tables
- Grant the CATALOG_READ permission
- Encrypted logs - To be able to read encrypted logs, Replicate should be able to access root key backups or the Replicate user should be granted the ENCRYPTION ROOT KEY ADMIN permission.

Supported data types

The following table shows the SAP HANA source data types that are supported when using Qlik Replicate, and the default mapping to Qlik Replicate data types.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

Data types

SAP HANA Data Types	Qlik Replicate Data Types
DATE	DATE
TIME	TIME
SECONDDATE	DATETIME
TIMESTAMP	DATETIME
TINYINT	UINT1
SMALLINT	INT2
INTEGER	INT4
BIGINT	INT8
DECIMAL (p,s)	NUMERIC
DECIMAL	STRING
SMALLDECIMAL	STRING
REAL	REAL4
DOUBLE	REAL8
VARCHAR	STRING
NVARCHAR	WSTRING
ALPHANUM	STRING
SHORTTEXT	WSTRING
VARBINARY	BYTES
BLOB	BLOB
CLOB	CLOB
NCLOB	NCLOB
TEXT	NCLOB
BOOLEAN	UINT1

Unsupported data types

The following data types are not supported:

- ARRAY
- ST_GEOMETRY
- ST_POINT

Setting general connection properties

You can add a SAP HANA endpoint to Qlik Replicate to use as a source. For more information on how to add endpoints, see *Defining and managing endpoints (page 114)*.

To add a SAP HANA source endpoint to Qlik Replicate:

1. In the Qlik Replicate console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, optionally enter a description that helps to identify the SAP HANA database.
4. Select **Source** as the database **role**.
You can do this step before any of the other steps if you want, however before you can continue with the next step in this process, you must select the database **role**.
5. Select **SAP HANA** as the database **Type**.



*When this endpoint is used as a duplicated source in a Log Stream Staging setup, select the **Read changes from log stream** check box and then select the relevant Log Stream Staging task from the drop-down list.
For information on setting up and managing Log Stream Staging tasks, see *Using the Log Stream (page 1072)*.*

6. In the **Server** field, specify the IP address or host name of the SAP HANA database server.
To connect to a High Availability Cluster, specify all of the cluster nodes and port numbers in the **Server** field. The nodes should be separated by a comma.
Example:
12.12.1.123:3033,12.12.1.124:3034
7. In the **Instance number** field, enter the instance number of the SAP HANA database with the source tables.
8. Select either **Single tenant** or **Multi tenant** according to your **Database architecture**. If you selected **Multi tenant**, enter the source **Database name**.
9. Enter the **Username** and **Password** required to access the SAP HANA database. If you do not know this information, see your SAP HANA database Administrator (DBA).



This information is case sensitive.

10. To connect to SAP HANA using SSL, in the **Security** section, do the following:
 - a. Select the **Enable SSL** check box.
 - b. In the **Crypto provider** field, select your cryptographic library provider from the drop-down list.

- c. In the **Keystore file** field, enter the file path to the SSL Keystore file that contains the client identity.
- d. In the **Truststore file** field, enter the file path to your SSL Truststore file that contains the server's public certificates.
- e. In the **Server host name in certificate** field, enter the name of the server host specified in the server certificate. This verifies the identity of the server, rather than the host name with which the connection was established. If you specify * as the host name, the server host name is not validated.
- f. In the **Server name indication** field, enter the name of the SSL server.
- g. To validate your SSL certificate, select the **Validate certificate** check box. This step is optional, but strongly recommended for security reasons.



To connect to SAP HANA using SSL, the client must be configured to authenticate the SSL server certificate. For more information, see Prerequisites (page 538).

11. Click **Test Connection** to verify that the specified settings are correct.

Setting advanced properties

In the **Advanced** tab, you can configure CDC-related options as well as any additional settings or internal parameters that may be required.

Select either **Trigger-based CDC** or **Log-based CDC** according to your preferred CDC method.



Switching between Trigger-based CDC and Log-based CDC (or vice versa) will require you to reload the target.

Trigger-Based CDC

- **Create CDC artifacts:** When this option is enabled (the default), Replicate creates the required CDC artifacts in the source database. Clear this check box if you would rather create the CDC artifacts manually.
- **Create CDC table in schema:** The name of the schema in the source database where you want the **attrep_cdc_changes** table to be created.



The specified schema must already exist in the source database.

- **Log cleanup interval (min):** Specify how often (in minutes) to check if Change Records need to be deleted from the **attrep_cdc_changestable**.
- **Log retention period (min):** Specify how long (in minutes) to keep Change Records in the **attrep_cdc_changes** table before deleting them.
- **Transactional consistency (sec):** By default, the longest supported transaction length is 60 seconds. If you have transactions in your environment that exceed 60 seconds, you should set this value to the

expected transaction duration. Note, however, that the longer the transaction, the greater the latency will be.

- **Use log table:** When this option is enabled, during the task, Replicate copies the changes from the **attrep_cdc_changes** table to the **attrep_cdc_log** table. This is the preferred method of working as it significantly reduces latency. In order to use this option, you must first perform the following procedure:

1. Stop all tasks configured with a SAP HANA source endpoint.
2. Back up the **attrep_cdc_changes** table.
3. In following script, replace `$$SCHEMA$` with the artifact's schema:

```
; Create the attrep_log_table

CREATE COLUMN TABLE "$SCHEMA$"."attrep_cdc_log"(

"CHANGE_ROWID" BIGINT CS_FIXED,

"CHANGE_INDX" BIGINT CS_FIXED,

"CHANGE_EVENT_TIME" LONGDATE CS_LONGDATE,

"COPY_TIME" LONGDATE CS_LONGDATE,

"OPER" CHAR(1) CS_FIXEDSTRING,

"TBL_DB_ID" BIGINT CS_FIXED,

"PK_0" NVARCHAR(5000),

"PK_1" NVARCHAR(5000),

"PK_2" NVARCHAR(5000),

"PK_3" NVARCHAR(5000),

"PK_4" NVARCHAR(5000),

"PK_5" NVARCHAR(5000),

"PK_6" NVARCHAR(5000),

"PK_7" NVARCHAR(5000),

"PK_8" NVARCHAR(5000),

"PK_9" NVARCHAR(5000),

"PK_10" NVARCHAR(5000),

"PK_11" NVARCHAR(5000),

"PK_12" NVARCHAR(5000),
```

```
"PK_13" NVARCHAR(5000),  
  
"PK_14" NVARCHAR(5000),  
  
"PK_15" NVARCHAR(5000),  
  
"PK_16" NVARCHAR(5000),  
  
"PK_17" NVARCHAR(5000),  
  
"PK_18" NVARCHAR(5000),  
  
"PK_19" NVARCHAR(5000),  
  
PRIMARY KEY ("CHANGE_ROWID");  
  
;Copy all changes from attrep_cdc_changes table to the attrep_cdc_log table  
INSERT INTO "$SCHEMA$"."attrep_cdc_log"  
("CHANGE_ROWID","CHANGE_INDX","CHANGE_EVENT_TIME","COPY_TIME","OPER","TBL_DB_ID",  
"PK_0", "PK_1", "PK_2", "PK_3", "PK_4", "PK_5", "PK_6", "PK_7", "PK_8", "PK_9",  
"PK_10", "PK_11", "PK_12", "PK_13", "PK_14", "PK_15", "PK_16", "PK_17", "PK_18",  
"PK_19")  
(SELECT "$rowid$","INDX","EVENT_TIME",CURRENT_TIMESTAMP,"OPER","TBL_DB_ID",  
"PK_0", "PK_1", "PK_2", "PK_3", "PK_4", "PK_5", "PK_6", "PK_7", "PK_8", "PK_9",  
"PK_10", "PK_11", "PK_12", "PK_13", "PK_14", "PK_15", "PK_16", "PK_17", "PK_18",  
"PK_19"  
FROM "$SCHEMA$"."attrep_cdc_changes" ORDER BY "$rowid$");  
  
;DELETE all changes from attrep_cdc_changes table  
DELETE FROM "$SCHEMA$"."attrep_cdc_changes" WHERE "$rowid$" IN (SELECT "CHANGE_  
ROWID" FROM "$SCHEMA$"."attrep_cdc_log");  
  
; COMMIT the changes  
  
COMMIT;
```

4. Run one script operation at a time with auto-commit off.



The duration of the script job will depend on size of the "attrep_cdc_changes" table.

5. Verify the following:
 - The **attrep_cdc_log** table has been created in the artifact's schema.
 - The **attrep_cdc_changes** table is empty.
 - The **attrep_cdc_changes** table content has been successfully copied to the new **attrep_cdc_log** table.
6. Resume your tasks.
 - **SAP archiving users to filter:** Specify the user names of SAP application users whose DELETE operations you do not want Replicate to capture. Users must be separated with commas.



Editing the user list requires you to stop and then resume the task for the changes to take affect.

Log-based CDC

Log-based CDC is faster than Trigger-based CDC and has a smaller system footprint. When **Log-based CDC** is selected, Replicate will read the changes directly from the SAP HANA logs. In order for Replicate to be able to do this, you need to specify the location of the logs in the **Redo Log Shared Folder** and **Backup Log Shared Folder** fields respectively (both fields are mandatory).

If Replicate Server is not installed on the SAP HANA machine, the redo logs and backup logs must reside in Shared Folders on the SAP HANA machine.

Log Access Preferences

You can determine which logs Replicate will read the changes from. If the logs are encrypted, you also need to provide the encryption settings as described in **Encryption Settings** below.

From the **Read changes from** drop-down list, select one of the following:

- **Prioritize backup log** - This is the default mode as it provides best performance while also offering greater efficiency. In this mode, Replicate will first try to capture the changes from the backup log. If the changes do not exist in the backup log, it will try to capture them from the online redo log instead. When this option is selected, you also need to specify the locations of the Redo log shared folder and the Backup log shared folder.
- **Backup log only** - In this mode, Replicate requires access to the backup log only. Changes will be captured from the backup log as soon as they become available. The backup interval and the size of the backup files will impact the capture time. When this option is selected, you also need to specify the location of the Backup log shared folder.
- **Redo log only** - This mode should not be used in production environments. In this mode, Replicate requires access to the redo log only. Changes will only be captured if they exist in the redo log. In the event that the changes are not available in the redo log, the task will need to be reloaded to ensure that all data and changes from the source are available on the target. When this option is selected, you also need to specify the location of the Redo log shared folder.

Encryption Settings

Replicate can decrypt encrypted redo logs and backup logs. The encryption root keys can either be retrieved automatically by Replicate or provided by the user in the designated fields.

- **Automatically retrieving the encryption root keys** - This method is preferable when the encryption root keys frequently change or if the encryption root key backup files are not accessible from the Replicate machine. This method also requires the Replicate user to be granted the ENCRYPTION_ROOT_KEY_ADMIN permission (for retrieving the encryption root keys).
- **Manually providing the encryption root keys** - This method is preferable when the encryption root keys rarely change and when the encryption root key backup files are accessible from the Replicate machine. This method may also be preferable for organizations that do not want to grant the ENCRYPTION_ROOT_KEY_ADMIN permission to the Replicate user.

The following settings are available:



Depending on your preferred method of providing Replicate with the encryption root keys (automatically or manually), some of the settings may not be relevant.

- **Encrypted redo log** - Select this option if the redo logs are encrypted. If you need to manually provide the encryption root keys, specify the path of the redo log encryption root key backup file in the **Log root key backup (.rkb) file** field. Otherwise, leave this field empty.
- **Encrypted backup log** - Select this option if the backup logs are encrypted. If you need to manually provide the encryption root keys, specify the path of the backup log encryption root key backup file in the **Backup root key backup (.rkb) file** field. Otherwise, leave this field empty.
- **Root key backup password** - The password for reading the root key backup files.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

Removing Replicate artifacts from the database

In order to capture changes, Qlik Replicate creates various artifacts in the SAP HANA database when the task starts. When the task completes, you can safely remove these artifacts.

For information on the Replicate artifacts, see *Capturing data changes (page 538)*

9.32 Using SAP ODP as a source

This section describes how to define SAP ODP as a source endpoint in a replication task.

In this section:

- *Prerequisites (page 550)*
- *Limitations and considerations (page 552)*
- *Required permissions (page 552)*
- *Supported data types (page 555)*
- *Setting general connection properties (page 556)*
- *Setting advanced properties (page 559)*

Prerequisites

The following section describes the prerequisites for working with the Qlik Replicate SAP ODP endpoint.

ODP API

The SAP ODP endpoint uses ODP API 2.0 which is available from the following SAP basis levels and above:

- PI_BASIS 730 SP 14 (part of SAP NetWeaver 7.30 SP 14)
- PI_BASIS 731 SP 16 (part of SAP NetWeaver 7.03 SP 16 and 7.31 SP 16)
- PI_BASIS 740 SP 11 (part of SAP NetWeaver 7.40 SP 11)
- SAP_BW 750 SP 0 (incl. former PI_BASIS packages)



SAP_BASIS and PI_BASIS are interchangeable. In newer systems, the component name is SAP_BASIS.

For more information, see SAP Note 1931427.

For information on how to release extractors so that they are available for the ODP API, see SAP Note 2232584.

SAP Java Connector

To work with the SAP ODP endpoint, you need to install SAP Java Connector on the Replicate server.

Installing the SAP Java Connector on Replicate for Windows

To install the SAP Java Connector:

1. Download the SAP Java Connector 3.x (64-bit) for Windows from the SAP Support Portal. To find out the exact SAP Java Connector version you need to support your SAP BW system, refer to the SAP BW Help.



The SAP Java Connector 3.1 running on Windows operating systems requires the Microsoft Visual Studio 2013 C/C++ runtime libraries to be installed on the system. See SAP Note 2786882 for details.

2. Install the SAP Java Connector.
3. Copy the **Sapjco3.jar** and **Sapjco3.dll** files from the SAP Java Connector installation folder to the following location under the Replicate installation directory:
{REPLICATE_INSTALL_DIR}\endpoint_srv\endpoints\SAPOdp.
4. Restart all of the Replicate services.



*If SAP Extractor is already installed, move the **Sapjco3.jar** and **Sapjco3.dll** files from <REPLICATE_INSTALL_DIR>\endpoint_srv\externals to <REPLICATE_INSTALL_DIR>\endpoint_srv\endpoints\SAPOdp and restart all of the Replicate services.*

Installing the SAP Java Connector on Replicate for Linux

To install the SAP Java Connector:

1. Download the SAP Java Connector 3.x (64-bit) for Linux from the SAP Support Portal.
To find out the exact SAP Java Connector version you need to support your SAP BW system, refer to the SAP BW Help.
2. Unzip the file you downloaded to a folder of your choice.
The folder should contain the *Sapjco3.jar* and *libsapjco3.so* files.
3. Copy these files to the following location under the Replicate installation directory:
<INSTALL_DIR>/endpoint_srv/endpoint_s/SAPOdp.
4. Restart all of the Replicate services.



If SAP Extractor is already installed, copy the **Sapjco3.jar** and **Sapjco3.dll** files from <REPLICATE_INSTALL_DIR>/endpoint_srv/externals to the following locations:

- <REPLICATE_INSTALL_DIR>/endpoint_srv/endpoint_s/SAPOdp
- <REPLICATE_INSTALL_DIR>/endpoint_srv/endpoint_s/SAP

Then restart all of the Replicate services as described in Replicate server procedures (page 53).



The SAP ODP endpoint will not be available for selection until the Java connector libraries have been installed and the Replicate service has been restarted.

STXL table support

To expose the table STXL for extraction using the SAP ODP endpoint, a custom extractor needs to be created in transaction RSO2 in SAP GUI. This extractor can then be called by the SAP ODP endpoint.

Setting up SLT on SAP

In order to replicate data, you must first define the parameters that SAP Landscape Transformation Replication Server will use to replicate data from a source system to a target system. You specify this information in a configuration. A configuration contains information about the source system, the target system, and the relevant connections.

You create a configuration in SAP Landscape Transformation Replication Server system. To create a configuration, open the SAP LT Replication Server Cockpit (transaction LTRC) and choose Create Configuration.

Creating a configuration involves the following steps:

1. In the **Specify General Data** tab, provide the following information:
 - **Configuration Name:** You must specify a name for the configuration. This name is used for the schema that is created automatically in the target system.
 - **Description (Optional):** Optionally, specify a description for the configuration.
2. In the **Specify Source System** tab, select the **RFC Connection** option and provide the following information:
 - **RFC Destination:** Enter **NONE**.
 - **Allow Multiple Usage:** Select this option.

3. In the **Specify Target System** tab, select the **Other** option and provide the following information:
 - **Scenario:** Select **Operational Data Provisioning (ODP)** from the drop-down list.
 - **Queue Alias:** Specify a name.
4. In the **Specify Transfer Settings** tab, specify transfer settings appropriate for your environment. Under **Job Options**, we recommend setting at least five Data Transfer, Initial Load, and Calculation Jobs.
5. In the **Review and Create** tab, review your settings and create the configuration.
6. Grant the *Required permissions* (page 552).

Once you have created a configuration, you can then proceed to configure the relevant settings in the **Advanced** tab. See *Setting advanced properties* (page 559) for details.

Limitations and considerations

When using SAP ODP as a source endpoint in a Replicate task, the following limitations apply:

- The "Start from timestamp" and "Start from position" options for running tasks are not supported.
- The [Allow unlimited LOB size](#) and [Optimize handling when LOB size is less than](#) LOB handling options are not supported.
- Source lookup is not supported.
- SAP ODP endpoint uses the metadata of the ODP node. Consequently, the target tables will be created with these ODP-provided columns.
- Hierarchy objects and columns are not supported
- Metadata changes are not supported. In order to make changes to the metadata of the ODP node, the task must be reloaded.
- When using the SAP ODP endpoint in a Log Stream configuration, Full Load replication is not supported.

Required permissions

To use SAP ODP as a source in a Qlik Replicate task, the following permissions are required:

- A SAP user with authorization to monitor and delete delta initializations in transaction ODQMON.
- A SAP user configured with the following authorization settings (as a minimum):

Function modules

Authorization object	RFC name	Activity
S_RFC	RFCPING	16
	RFC_FUNCTION_SEARCH	16
	RFC_GET_FUNCTION_INTERFACE	16
	RFC_METADATA_GET	16
	RODPS_REPL_CONTEXT_GET_LIST	16

9 Adding and managing source endpoints

RODPS_REPL_ODP_FETCH	16
RODPS_REPL_ODP_GET_DETAIL	16
RODPS_REPL_ODP_GET_LIST	16
RODPS_REPL_ODP_OPEN	16
RODPS_REPL_ODP_READ_DIRECT	16
RODPS_REPL_ODP_RESET	16
SADT_REST_RFC_ENDPOINT	16
SCSI_GET_SYSTEM_INFO	16
RFC1 (Only required when replicating from an SLT Replication Server)	16

Data objects

Authorization object	Field name	Value	Activity
S_RS_ADSO	RSINFOAREA	*	23
	RSOADSONM	*	
S_RS_COMP	RSINFOAREA	*	23
	RSINFOCUBE	*	
	RSZCOMPTP	*	
	RSZCOMPID	*	
S_RS_COMP1	RSZCOMPID	*	23
	RSZCOMPTP	*	
	RSZOWNER	*	
S_RS_DS	RSDS	*	23
	RSDSPART	*	
	RSLOGSYS	*	
S_RS_HCPR	RSINFOAREA	*	23
	RSHCPR	*	
	RSHCPROBJ	*	
S_RS_HYBR	RSHYBRPROV	*	23
	RSHYBROBJ	*	
S_RS_ICUBE	RSINFOAREA	*	23
	RSINFOCUBE	*	
	RSICUBEOBJ	*	

9 Adding and managing source endpoints

S_RS_IOBJ	RSIOBJ	*	23
	RSIOBJCAT	*	
	RSIOBJPART	*	
S_RS_IOMAD	RSINFOAREA	*	23
	RSAPPLNM	*	
	RSIOBJNM	*	
S_RS_ISET	RSINFOAREA	*	23
	RSINFOSET	*	
	RSISETOBJ	*	
S_RS_MPRO	RSINFOAREA	*	23
	RSMPRO	*	
	RSMPROOBJ	*	
S_RS_ODSO	RSINFOAREA	*	23
	RSODSOBJ	*	
	RSODSPART	*	
S_RS_ODP_H	RSODPHNAME	*	23
	RSODPHPKG	*	
S_RS_ODSV	RSFBPNAME	*	23
	RSFBPOBJ	*	
	RSINFOAREA	*	
S_RO_OSOA	OLTPSOURCE	*	23
	OSOAAPCO	*	
	OSOAPART	*	

Other

Authorization object	Field name	Value	Activity
S_ADMI_FCD	S_ADMI_FCD	PADM, ST22	
S_ADT_RES	URI	/sap/bc/adt/*	
S_DEVELOP	OBJTYPE	DEBUG, ST22	03
S_TCODE	TCD	ST22	
S_BTCH_ADM	BTCADMIN	Y	
S_BTCH_JOB	JOBACTION	RELE	
	JOBGROUP	*	

9 Adding and managing source endpoints

S_BTCH_NAM	BTCUNAME	BWREMOTE	
S_DMIS (Only required when replicating from an SLT Replication Server)	MBT_PR_ARE	SLOP	03
	MBT_PR_LEV	PACKAGE	

Supported data types

The table below shows the SAP ODP source data types that are supported when using Qlik Replicate and the default mapping to the Qlik Replicate data types.

For information on how the data type is mapped to the target, see the chapter for the target endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).

Data types

ABAP Data Type	Qlik Replicate Data Type
ACCP	DATE
CHAR	STRING
CLNT	STRING
CUKY	STRING
CURR	NUMERIC
DATS	DATE
DEC	NUMERIC
FLTP	REAL4
INT1	INT1
INT2	INT2
INT4	INT4
INT8	INT8
LANG	STRING
NUMC	STRING
QUAN	NUMERIC
RAW	STRING
SSTR	STRING
STRG	CLOB
TIMS	TIME
UNIT	STRING
VARC	STRING

Setting general connection properties

This section describes how to set up connection parameters for a specific SAP ODP server or for a SAP ODP system using load balancing.

Connecting to a specific SAP ODP Application server

To connect to a specific SAP ODP Application server:

1. In the Qlik Replicate console, click the **Manage Endpoint Connections** toolbar button to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button.
2. In the **Name** field, enter a display name for your endpoint.
3. Optionally, in the **Description** field, enter a description for the SAP Application endpoint.
4. Select **Source** as the database **role**.
5. Select **SAP ODP** as the database **Type**.
6. From the **Connection mode** drop-down list, select **Custom Application Server**.
7. In the **Server name** field, enter the IP address of the Application Server on which the SAP ODP source is located.
8. In the **Instance identifier** field, enter the instance number of the SAP ODP source you want to replicate.
9. In the **Client identifier** field, enter the System ID of the SAP ODP source you want to replicate.
10. Enter your credentials (**User Name**, **Password**) for accessing the SAP ODP source.
These are the credentials for the communication user created earlier in SAP.

Connecting to a SAP ODP system using load balancing

To connect to a SAP ODP system using load balancing:

1. In the Qlik Replicate console, click the **Manage Endpoint Connections** toolbar button to open the **Manage Endpoints Connections** dialog box.
2. In the **Name** field, enter a display name for your endpoint.
3. Optionally, in the **Description** field, enter a description for the SAP Application endpoint.
4. Select **Source** as the database **role**.
5. Select **SAP ODP** as the database **Type**.
6. From the **Connection mode** drop-down list, select **Load Balancing Server**.
7. In the **Message server** field, enter the host name or IP address of the message server host.
8. In the **Application servers group name** field, enter the name of the SAP server group. This is an optional group of application servers in a load balancing connection.
9. In the **SAP system name** field, enter the SAP R/3 name.
10. In the **Message server service** field, enter the name of the SAP message server service as specified in the following file:
<system drive>:\WINDOWS\system32\drivers\etc\services
If you do not specify a value, the Data Provider for SAP uses the following default name:
sapms<R/3 system name>
11. In the **Client identifier** field, enter the System ID of the SAP Application source you want to replicate.

12. Enter your credentials (**Username, Password**) for accessing the SAP ODP source. These are the credentials for the communication user created earlier in SAP.

Security

In the **Security** settings, you can configure Secure Network Communication (SNC).

Prerequisites for working with SNC

Follow the steps below to install the Secure Network Communication (SNC) client on the Replicate machine.

What you need:

- An exported certificate (.crt) of the SAP server
- SAPCAR.EXE
- SAP user (authorized customer)
- The version of the crypto library which is installed on the corresponding SAP server

Installing the SNC client

1. Create a workspace folder for the SAP SNC files and binaries (hereafter referred to as "your SNC folder"), for example: "C:\snc\"
2. Copy the exported server certificate and **SAPCAR.EXE** to your SNC folder.
3. Go to <https://support.sap.com/en/my-support/software-downloads.html> and search for **SAPCRYPTOLIB** under **Installations & Upgrades**. Download the 64-bit **.SAR** to your SNC folder.
4. Open a command prompt and change the working directory to your SNC folder. Then run the following command to unpack the content of the **.SAR** to your SNC folder:
`sapcar -xvf LibName.sar`
Example:
`sapcar -xvf SAPCRYPTOLIBP_8541-20011731_32.SAR`
5. Add system environment variables as follows:
 - a. Add a system environment called **SECUDIR** with the path to your SNC folder as its value.
 - b. Add a system environment variable called **QLIK_SNC_LIB** with the path to the **sapcrypto.dll** file as its value.
 - c. Add the newly added environment variables to the "PATH" environment variable.
6. Determine the **<PSE_File_Name>** and choose a **<PSE_PIN>** to protect it. You will need to provide this information in the next steps.
Example:
`pseName: "CN=USR,OU=SAP,O=Qlik,C=IS" password: password123`
7. Determine the **<SNC_NAME>**. It should look something like this: `CN=USR, OU=SAP, O=Qlik, C=IS`
See also *Determining the server SNC name (page 558)* below.
8. Make sure you have the required permissions to access and execute the files in the **SECUDIR** folder, and then run the following command to generate the PSE file:
`sapgenpse get_pse -p <PSE_File_Name>.pse -x <PSE_PIN> <SNC_NAME>`
Example:
`sapgenpse get_pse -p usr.pse -x password123 "CN=USR,OU=SAP,O=Qlik,C=IS"`

9. Bind the PSE file with the OS user and create the **CRED_V2** file in **SECUDIR** folder as follows:
 - a. Make a note of the OS user under which Replicate is running:
 - Windows: Open the Services console and double-click the QlikReplicate Server service to open the properties dialog. Look in the **Log On** tab.
 - Linux: Run the `ps aux` command.
 - b. Run the following command:

```
sapgenpse seclogin -p <PSE_File_Name>.pse -x <PSE_PIN> -O <OS_USER>
```

Example:

```
sapgenpse seclogin -p usr.pse -x password123 -O SYSTEM
```
10. Generate the CRT file by executing the following command:

```
sapgenpse export_own_cert -o <PSE_File_Name>.crt -p <PSE_File_Name>.pse -x <PSE_PIN>
```

Example:

```
sapgenpse export_own_cert -o usr.crt -p usr.pse -x password123
```
11. Import the SAP Application Server Certificate (<SERVER_CERT>) to the PSE by executing the following command:

```
sapgenpse maintain_pk -a <SERVER_CERT>.crt -p <PSE_File_Name>.pse -x <PSE_PIN>
```

Example:



```
sapgenpse maintain_pk -a sapsys.crt -p usr.pse -x password123
```
12. To verify that the DN of the SAP Server's PSE was imported into the client, run the following command and then check the "subject" value:

```
sapgenpse maintain_pk -v -l -p <PSE_File_Name>.pse
```

Example:

```
sapgenpse maintain_pk -v -l -p usr.pse
```

Importing the client certificate

1. Connect to the SAP Application Server and navigate to the "STRUST" transaction using an authorized user.
2. Double-click the **SNC (SAPCryptolib)** folder.
3. Click  to switch to Change view.
4. Click  to import the certificate.
5. In the new dialog, enter the path to the .crt file that was created earlier, then click continue.
6. Verify the details of the certificate in the **Certificate** section.
7. Click **Add to Certificate List** to add the certificate to the list.
8. Save the changes.

Determining the server SNC name

There are two ways you can determine the server name:

- **Method 1:** Decrypt the server CRT file using the OpenSSL command. The server name will be part of the subject.
- **Method 2:** This method requires appropriate permissions. While connected to the system:
 1. Run the RZ10 transaction.
 2. Select the system profile.

3. Select the **Extended Maintenance** option and then click **Display**.
4. The value of the `snc/identity/as` parameter should be the SNC name.

Connection settings

Configure the SNC settings in the SAP ODP endpoint as follows:

- **Activate Secure Network Communication:** Select to turn on SNC.
- **SNC name:** The SNC partner name.
Example:
`p:CN=SYS, OU=SAP, O=Qlik, C=IS`
- **SNC quality of protection** - Select one of the following:
 - **Authentication only:** Select to verify the identity of the SAP ODP machine. This is the minimum protection level offered by SNC.
 - **Integrity protection:** Select to detect any changes or manipulation of the data, which might have occurred between the Replicate machine and the SAP ODP machine.
 - **Privacy protection:** Select to encrypt the messages being transferred to prevent eavesdropping. Privacy protection also includes integrity protection. This is the maximum level of protection provided by SNC.
 - **Maximum security available:** The maximum level of data protection supported by the SAP ODP machine.

Setting advanced properties

In the **Advanced** tab, you can set the following parameters:

- **Subscriber type** - Choose one of the following:
 - **SAP NetWeaver Business Warehouse** - This is the default ODP subscriber type.
 - **RODPS_REPL_TEST** - Used mainly for testing and is not recommended to use for production.
 - **Others** - This will expose the field **Other subscriber type**, where you can specify a subscriber type that is not listed.
- **Subscriber name** - Specify the subscriber name. This will be the name shown in SAP for all subscriptions opened by this endpoint connection.
- **ODP context** - Choose one of the following:
 - **ABAP Core Data Services (CDS) views** - The context of the CDS views. This is the default ODP context.
 - **SAP NetWeaver Business Warehouse** - The context of the BW objects.
 - **SAP DataSources/Extractors** - The context of the data sources and extractors.
 - **SAP HANA Information Views** - The context of the HANA views.
 - **SAP LT Replication (SLT)** - Required to work with a SAP Landscape Transformation Replication Server.
See also: [Setting up SLT on SAP \(page 551\)](#)
- **SLT alias** - This field is only visible when the **ODP context** is set to **SAP LT Replication (SLT)**. Specify the SLT alias as defined in the configuration created in *[Setting up SLT on SAP \(page 551\)](#)*
- **Delta extraction interval** - Defines how frequently the deltas are extracted from your system.

- The default value is 60 minutes.
- Setting the value to 0 means that all the datasets of the task will run only one delta/initial load and then the task will stop automatically.
- **Max package size** - The size of the data package in bytes. This value can be adjusted according to your network and system capabilities.

Delta processing settings

- **Delta processing mode** - Select one of the following:
 - **History data** - This is the default mode. When this mode is selected, all data will be applied as INSERTS, thereby preserving previous record versions. When working in this mode, the following options are also available:
 - **Apply original primary key** - Retrieve the primary key settings from the ODP metadata.
 - **Reverse summable fields** - Reverse the values of summable fields. Before-image values have reversed signs. Enabling this setting will restore the sign to its prior state and recalculate the original value.
 - **Current data** - When this mode is selected, the actual change operation (INSERT, UPDATE, or DELETE) will be performed on the target.



- *Changes operations are determined by the ODP source. Some operations might not be supported on the target.*
- *We recommend manually setting a Primary Key for those ODP sources that are not set up to use the default Primary Key.*
- *Some UPDATES are received as INSERTs. Therefore, to prevent conflicts on the target, set the **Duplicate key when applying an INSERT** Apply Conflicts handling policy to **UPDATE the existing target record**. For more information on this setting, see *Apply Conflicts* (page 1191).*

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

9 Adding and managing source endpoints

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

9.33 Using SAP Sybase ASE as a source

This section describes how to set up and use a SAP Sybase ASE database as the source endpoint in a replication task.

In this section:

- *Prerequisites (page 563)*
- *Limitations and considerations (page 563)*
- *Required permissions (page 564)*
- *SAP Sybase ASE database source data types (page 564)*
- *Setting general connection properties (page 566)*
- *Setting advanced connection properties (page 567)*
- *Handling the secondary truncation point (page 569)*

Prerequisites

This topic lists the prerequisites for using the SAP Sybase ASE source endpoint in a replication task.

General prerequisites

- SAP Sybase ASE replication enabled for tables using the `sp_setreptable` command or privileges to enable it automatically.
- RepAgent must be disabled on the SAP Sybase ASE database.
- When replicating to SAP Sybase ASE 15.7 installed on a Windows machine configured with a non-Latin language (e.g. Chinese), Qlik Replicate requires Sybase 15.7 SP121 to be installed on the SAP Sybase ASE machine.
- Turn off automatic truncation by running the following command:

```
sp_dboption mydb, "trunc log on chkpt", false  
go
```

Replicate on Windows

Install SAP Adaptive Server Enterprise ODBC Driver 16.0 (64-bit) on the Qlik Replicate computer.

Replicate on Linux

1. Install SAP Adaptive Server Enterprise ODBC Driver 16.0 (64-bit) on the Qlik Replicate computer.
2. After the driver is installed, verify that the `/etc/odbcinst.ini` file contains an entry for Sybase, as in the following example:

```
[Adaptive Server Enterprise]  
Description=Sybase ODBC Driver  
Driver=/opt/sap/DataAccess64/ODBC/lib/libsybdrvodb.so  
Setup=  
FileUsage=-1  
CPOutput=  
CPReuse=
```
3. Verify that the symbolic link refers to `libsybdrvodb-sql1en8.so`.
In the event that it does not, redirect the symbolic link to `libsybdrvodb-sql1en8.so` by running the following command:

```
ln -fs ./libsybdrvodb-sql1en8.so libsybdrvodb.so
```

Limitations and considerations

The following limitations apply:

- Sybase ASE primary/standby configuration is supported with the publish-and-subscribe model only (i.e. Warm standby/MSA is not supported).
- Only one Qlik Replicate task can be run per SAP Sybase ASE database.
- Rename table is not supported (e.g. `sp_rename 'Sales.SalesRegion', 'SalesReg';`)
- Rename column is not supported (e.g. `sp_rename 'Sales.Sales.Region', 'RegID', 'COLUMN';`)

- Zero values located at the end of binary data type strings are truncated when replicated to the target database. For example, 0x00000000000000000000000010000010000000 in the source table will become 0x0000000000000000000000001000001 in the target table.
- The reorg rebuild index command is not supported.
- Clusters are not supported.
- Materialized views are not supported.
- Data capture from partitions is only supported from Sybase ASE 15 or later.
- Merge operations performed during CDC will result in a non-recoverable error.
- The WRITETEXT statement is not supported as it may result in duplicate (or more) inserts on the target.
- The Replicate AR_H_USER header column is not supported for database changes performed in auto-commit mode. For these changes, the header column will be empty.
- Rollback trigger on datarow locked tables is not supported.
- Replicating data with 4-byte emoji characters is not supported.

Required permissions

To use SAP Sybase ASE database as a source in a Replicate task, the following permissions are required:

- sa_role
- replication_role
- sybase_ts_role

If the **Automatically enable Sybase replication** option is enabled (in the **Advanced** tab), Replicate also needs permission to run the stored procedure sp_setreptable.

For information on the **Automatically enable SAP Sybase ASE replication** option, see *Setting advanced connection properties (page 567)*.

SAP Sybase ASE database source data types

The following table shows the SAP Sybase ASE database source data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.

For information on how to view the data type that is mapped in the target, see the section for the target database you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

Data types

SAP Sybase ASE Source Data Types	Qlik Replicate Data Types
BIGINT	INT8
UNSIGNED BIGINT	UINT8
INT	INT4
UNSIGNED INT	UINT4

9 Adding and managing source endpoints

SAP Sybase ASE Source Data Types	Qlik Replicate Data Types
SMALLINT	INT2
UNSIGNED SMALLINT	UINT2
TINYINT	UINT1
DECIMAL	NUMERIC
NUMERIC	NUMERIC
FLOAT	REAL8
DOUBLE	REAL8
REAL	REAL4
MONEY	NUMERIC
SMALLMONEY	NUMERIC
DATETIME	DATETIME
BIGDATETIME	DATETIME (6)
SMALLDATETIME	DATETIME
DATE	DATE
TIME	TIME
BIGTIME	TIME
CHAR	STRING
UNICHAR	WSTRING
NCHAR	WSTRING
VARCHAR	STRING
UNIVARCHAR	WSTRING
NVARCHAR	WSTRING
BINARY	BYTES
VARBINARY	BYTES
BIT	BOOLEAN
TEXT	CLOB
UNITEXT	NCLOB
IMAGE	BLOB

Non-supported data types

Source SAP Sybase ASE tables with columns of the following SAP Sybase ASE data types cannot be replicated. Replicated columns with these data types will show as null.

- UDT

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.



You can also use SAP Sybase ASE files as a source. For more information, see [Using the Qlik Replicate file channel \(page 1061\)](#).

To add a SAP Sybase ASE source endpoint to Qlik Replicate:

1. In the Qlik Replicate Console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see [Defining and managing endpoints \(page 114\)](#).
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the SAP Sybase ASE database. This is optional.
4. Select **Source** as the endpoint **role**.
5. Select **SAP Sybase ASE** as the database **Type**.



*When this endpoint is used as a duplicated source in a Log Stream Staging setup, select the **Read changes from log stream** check box and then select the relevant Log Stream Staging task from the drop-down list.
For information on setting up and managing Log Stream Staging tasks, see [Using the Log Stream \(page 1072\)](#).*

6. In the **Server Name** field, enter the host name or IP address of the computer on which the SAP Sybase ASE database is installed.



Consider the following:

- This information is case sensitive.
- You can use the **Advanced** tab to add specific properties and create a custom connect string. In this case, you do not need to enter information in this tab. For more information on using the **Advanced** tab, see *Setting advanced connection properties* (page 567).
- To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**. If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box. To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

7. Optionally, change the default port (5000).
8. Type the SAP Sybase ASE authentication information (**User Name**, **Password**) for the authorized user for this SAP Sybase ASE database. If you do not know this information, see your SAP Sybase ASE database Administrator (DBA).



Consider the following:

- This information is case sensitive.
- This information is required. If you are using the **Advanced** tab to create a custom string, make sure to include the **User Name** and **Password** properties. See *Setting advanced connection properties* (page 567) for more information.
- If you want to set custom properties for this database, see *Setting advanced connection properties* (page 567).



Make sure that the SAP Sybase ASE user entered in the SAP Sybase ASE Authentication section has the correct access privileges. For information on how to provide the required privileges, see *Required permissions* (page 564).

9. In the **Database name** field, enter the SAP Sybase ASE database name.

Setting advanced connection properties

In the **Advanced** tab, you can set the following parameters:

- **Automatically enable SAP Sybase ASE replication:** Select this to automatically enable SAP Sybase ASE replication. This is only required if SAP Sybase ASE replication has not been enabled already. For more information, see *Prerequisites* (page 563).
- **Additional ODBC connection properties:** Specify any additional ODBC connection parameters that you want to use.



If the user name or password specified in the **General** tab contains non-Latin characters (e.g. Chinese), the following property is required:
`charset=gb18030`

- **Advance truncation point every (seconds):** The frequency with which to advance the secondary truncation point when reading changes from the Sybase ASE log.
- **Dump the log every (seconds):** How often to dump the parts of the log which Replicate has finished reading (i.e. the log behind the secondary truncation point). The default is "0" which means that Replicate never dumps the log.



You must change the default value. If you do not change the default value, the log will eventually consume all available disk space. In addition to changing the default value, you also need to create a stored procedure and turn off automatic truncation on checkpoint as described below.

1. Create a stored procedure for dumping the log, as shown in the following example:

```
create procedure attrep_dumpaction @DbName varchar(255)
as
if @DbName = '<source_db>'
Begin
<Check dump log not running.>
<Add your database logic here.>
print "Truncating log for database '%1!'", @DbName
dump transaction <source_db> with truncate_only
end
```

2. Turn off automatic truncation on checkpoint as follows:

```
sp_dboption <source_db>, "trunc log on chkpt", false
go
```

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

Handling the secondary truncation point

When a task starts, Replicate establishes a `$replication_truncation_point` entry in the `syslogshold` system view, indicating that a replication process is in progress. While Qlik Replicate is working, it advances the replication truncation point at regular intervals, according to the amount of data that has already been copied to the target.

Once the `$replication_truncation_point` entry has been established, the Replicate task must be kept running at all times to prevent the database log from becoming excessively large. If you want to stop the Replicate task permanently, the replication truncation point must be removed by issuing the following command:

```
dbcc settrunc('ltm','ignore')
```

After the truncation point has been removed, the Replicate task cannot be resumed. The log will continue to be truncated automatically at the checkpoints (if automatic truncation is set).



Qlik Replicate requires the secondary truncation point to be valid. Manually changing the secondary truncation point may result in changes not being captured or the task being stopped due to errors.

9.34 Using Teradata Database as a source

This section describes how to set up and use Teradata Database as a source in a replication task.

In this section:

- *Prerequisites (page 571)*
- *Limitations and considerations (page 572)*
- *Required permissions (page 572)*
- *Teradata source data types (page 572)*
- *Setting general connection properties (page 573)*
- *Setting Change Processing parameters (page 574)*

Prerequisites

The following section describes the prerequisites for working with Qlik Replicate and a Teradata Database Source.

Replicate Server for Windows

Install Teradata Database ODBC Driver 17.20 for Windows.

Download the driver from <https://downloads.teradata.com/download/connectivity/odbc-driver/windows>

Replicate Server for Linux

The following section describes the steps you need to perform to work with Qlik Replicate for Linux and Teradata Database as a source endpoint in a Replicate task. Teradata Database Client requires the DataDirect ODBC driver manager (provided with Teradata Database Client).



These instructions assume that Teradata Database Client 17.20 is being installed in the following location:

`/opt/teradata/client/17.20`

If you are installing another version or to a different location, simply change the instructions accordingly.

1. Stop the Replicate service:
`./areplicate stop`
2. Optionally, confirm that the service has stopped:
`./areplicate status`
3. Install Teradata Database Client 17.20 or later for Linux.
Download the driver from <https://downloads.teradata.com/download/connectivity/odbc-driver/linux>
4. Run the following commands:
`TD_CLIENT=/opt/teradata/client/17.20/odbc_64/lib:/opt/teradata/client/17.20/lib64; export TD_CLIENT`
`ODBCINI=/opt/teradata/client/17.20/odbc_64/odbc.ini; export ODBCINI`
`ODBCINST=/opt/teradata/client/17.20/odbc_64/odbcinst.ini; export ODBCINST`
5. Run the following command:
`LD_LIBRARY_PATH=$TD_CLIENT:/opt/attunity/replicate/lib:/lib64:/usr/lib64; export LD_LIBRARY_PATH`
6. To verify that the path is updated, run the following command:
`echo $LD_LIBRARY_PATH`
7. Run the following command:
`export ODBCINI=/opt/teradata/client/17.20/odbc_64/odbc.ini`
8. Add the export commands to the `site_arep_login.sh` file.
9. Start the Replicate service:
`./areplicate start`
10. Optionally confirm that the service has started:
`./areplicate status`



A Replicate task cannot be defined with endpoints that use different ODBC Driver Managers. Teradata Database source is accessed using the *DataDirect* ODBC Driver Manager while most target endpoints (that can be accessed by Replicate running on Linux) are accessed using the *unixODBC* Driver Manager.

Consequently, when replicating to all target endpoints excluding Oracle, Hadoop, and File (which do not use the *unixODBC* Driver Manager), you need to use the Replicate File Channel. For more information about setting up a task using the File Channel, see *Using the Qlik Replicate file channel* (page 1061).

Limitations and considerations

Replicating data with 4-byte emoji characters is not supported.

Required permissions

The user that is specified in the **General** tab when *Setting general connection properties* (page 573) must be registered as a user in the Teradata Database.

Teradata source data types

The following table shows the Teradata source data types that are supported when using Qlik Replicate and the default mapping to the Qlik Replicate data types.

For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).

Teradata and Qlik Replicate data types

Teradata Data Types	Qlik Replicate Data Types
BLOB	BLOB
BYTE	BYTES
BYTEINT	INT1
BIGINT	INT8
DATE	DATE
DECIMAL	REAL8
DOUBLE PRECISION	REAL8
FLOAT	REAL8
INTEGER	INT4
INTERVAL DAY	STRING (Support a maximum of 9,999 days)
INTERVAL DAY TO HOUR	STRING
INTERVAL DAY TO MINUTE	STRING
INTERVAL DAY TO SECOND	STRING

Teradata Data Types	Qlik Replicate Data Types
INTERVAL HOUR	STRING
INTERVAL HOUR TO MINUTE	STRING
INTERVAL HOUR TO SECOND	STRING
INTERVAL MINUTE	STRING
INTERVAL MINUTE TO SECOND	STRING
INTERVAL SECOND	STRING (Supports up to six fractional seconds)
CHAR	STRING
CLOB	CLOB
GRAPHIC	STRING
INTERVAL MONTH	STRING
INTERVAL YEAR	STRING
INTERVAL YEAR TO MONTH	STRING
REAL	REAL8
SMALLINT	INT2
TIME	TIME
TIMESTAMP	DATETIME
TIMESTAMP WITH TIME ZONE	DATETIME
TIME WITH TIME ZONE	TIME
VARBYTE	BYTES
VARCHAR	STRING (10)
VARGRAPHIC	STRING (10)
NUMERIC	If scale is => 0 and <=37, then: NUMERIC (p,s) If scale is => 38, then: VARCHAR (45)
CHAR VARYING	STRING
LONG VARCHAR	STRING

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure change processing parameters, see [Configuring Change Processing](#) below.

To add a Teradata Database source endpoint to Qlik Replicate:

1. In the Qlik Replicate console, click **Manage Endpoint Connections** to open the **Manage Endpoint Connections** dialog box. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints* (page 114).
2. In the **Name** field, type a name for your Teradata database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Teradata database. This is optional.
4. Select **Source** as the database **role**.
5. Select **Teradata Database** as the database **Type**.



*When this endpoint is used as a duplicated source in a Log Stream Staging setup, select the **Read changes from log stream** check box and then select the relevant Log Stream Staging task from the drop-down list.*

*For information on setting up and managing Log Stream Staging tasks, see *Using the Log Stream* (page 1072).*

6. Type the **Server** name. This is the name of the computer with the Teradata Database instance you want to work with.
7. Type the Teradata Database authentication information (**Username, Password**) for the authorized user for this Teradata Database. If you do not know this information, see your Teradata Database system manager.



Consider the following:

- *This information is case sensitive.*
 - *To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**. If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.*
- To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.*



*Make sure that the Teradata Database user entered in the Teradata Database Authentication section has the correct access privileges. For information on how to provide the required privileges, see *Required permissions* (page 572).*

8. Type the **Default database** name or select one from the list of available endpoints. This is the name of the Teradata Database where you are replicating the data to.

Setting Change Processing parameters

The **Change Processing** tab lets you define change processing settings for the Teradata Database source. Normally, Replicate scans a database's transaction logs for changes and then applies those changes to the target database. However, this method of change processing is not possible with Data Warehouse endpoints

such as Teradata Database since these endpoints do not generate transaction logs.

The good news is that you can still use Replicate to capture changes from Teradata Database - it just requires a little bit of preparation.

Prerequisites

Before you can define the settings in the **Change Processing** tab, you need to ensure that at least one special "Context" column exists in your source database tables. Context column(s) are basically columns in a table that enable Replicate to determine whether the data has changed. You can add Context columns specifically for the purpose of change processing (either using a script or manually) or you can use existing columns that contain suitable "Context" data.



You can create and reference any number of Context columns in a table as long as the Context column names are the same for all source tables. Additionally, each value in the Context column(s) must be unique.



When working with non-datetime Context columns such as integers that are not based on epoch time (for example, counters, strings, and numeric values), the values of those columns should be created by a singleton entity. This is required to ensure that the results are returned in the correct sequence (thereby preventing loss of data) when the Context column values are generated by multiple concurrent processes. It is strongly recommended not to generate Context column values from multiple concurrent sources as doing so poses a high risk of losing events.

In the example below, the Context column **cf** has been added to the table. The **cf** column contains TIMESTAMPS that enable Replicate to determine whether a change occurred (by comparing the current TIMESTAMP with the TIMESTAMP stored in its repository).

By default, all changes are assumed to be INSERTS. If UPDATE and DELETE operations are also performed on the source tables, you can write an UPDATE and/or DELETE expression (described below) that will enable Replicate to identify the operation type.

a	first_name	cf ▼	oper
1	may1	2014-12-29 14:52:2...	D
2	may2	2014-12-29 14:52:2...	D
6	omri6	2014-12-29 14:53:5...	D
8	omri8	2014-12-29 14:53:5...	D
9	omri9	2014-12-29 14:53:5...	U
10	omri10	2014-12-29 14:53:5...	U
11	omri11	2014-12-29 14:53:5...	U
3	pitz	2014-12-30 11:04:0...	U
7	gal	2014-12-30 13:27:1...	U
5	gal	2014-12-30 13:29:2...	U
12	gal	2014-12-30 13:41:5...	U

Limitations

The following limitations apply when Change Processing is enabled for the Teradata Database source:

- The "Start from timestamp" run option is not supported. For more information, see *Advanced Run Options (page 1205)*.
- If one of the Context columns is part of the Primary Key or Unique Index, then UPDATE and DELETE operations are not supported.
- Context columns cannot be LOB columns
- DDLs are not supported
- When inserting a record and then updating the same record, the task error handling settings should be set as follows:
 1. Open the **<Task Name> Settings** dialog box.
 2. Select the **Error Handling|Apply Conflicts** tab.
 3. Set a task-specific Apply Conflicts policy as described in *Error Handling settings (page 1187)*.
 4. From the **No record found for applying an update** drop-down list, select **INSERT the missing target record**.

For more information on error handling, see *Error handling (page 1186)*.

Configuring Change Processing settings

Perform the following steps to configure change processing settings.

To configure change processing settings:

1. Select the **Change Processing** tab in the Teradata Database source.
2. In the **Columns** field, specify the names of the Context columns. The column names are case-sensitive and must be separated by commas.

Example:

context1,context2

3. Choose the sorting order of the Context columns as appropriate (**Ascending** or **Descending**). Note that if the order you select is not the same as the actual sorting order, an error will occur.
4. In the **Check for changes every** field, specify how often to check for changes.
5. Enter expressions that Replicate will use to identify UPDATE and DELETE operations. If you do not enter any expressions or if no match is found for an expression, any row whose context is higher (if the sorting order is **Ascending**) or lower (if the sorting order is **Descending**) than the previous context value will be considered an INSERT.



Expressions must be written in the native syntax of the Teradata Database source. All examples in this section are written using PostgreSQL syntax.

- **Update expression** - Enter an expression for identifying UPDATE operations.

Example (based on [Figure "Example of a Table with a Context Column"](#)):

```
case when oper='U' then 1 else 0 end
```



*[Selecting the **UPDATE the existing target record** option in the Apply Conflicts (page 1191) tab, eliminates the need to provide an UPDATE expression.*

- **Delete expression** - Enter an expression for identifying DELETE operations.

Example (based on [Figure "Example of a Table with a Context Column"](#)):

```
case when oper='D' then 1 else 0 end
```



In addition to the DELETE expression, DELETE operations should be carried out as "Soft" deletes. This means that the row is not actually deleted from the table, but rather, marked as "deleted".

6. Select **Override connection string parameters** to append the connection string with parameters that are not exposed in the UI. As such parameters are normally not required, they should only be used after consulting with Qlik Support.

Additional settings and summary

In the **Change Processing** tab, you can also set internal parameters, set more options, and view a summary of your settings.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10 Adding and managing target endpoints

This topic describes how to configure target endpoint settings. When you define a replication task, the target endpoint is the data repository to which you wish to replicate your data.

In this section:

- *Using Actian Vector as a target (page 581)*
- *Using Amazon MSK as a target (page 588)*
- *Using Amazon EMR as a target (page 601)*
- *Using Amazon Kinesis data streams as a target (page 609)*
- *Using Amazon Redshift as a target (page 625)*
- *Using Amazon S3 as a target (page 634)*
- *Using Cloudera Data Platform (CDP) Private Cloud as a target (page 649)*
- *Using Confluent Cloud as a target (page 660)*
- *Using Databricks Lakehouse (Delta) as a target (page 671)*
- *Using Databricks (Cloud Storage) as a target (page 682)*
- *Using a file as a target (page 697)*
- *Using Google Cloud AlloyDB for PostgreSQL as a target (page 709)*
- *Using Google Cloud BigQuery as a target (page 715)*
- *Using Google Cloud Pub/Sub as a target (page 722)*
- *Using Google Cloud SQL for MySQL as a Target (page 735)*
- *Using Google Cloud SQL for PostgreSQL as a target (page 743)*
- *Using Google Cloud SQL for SQL Server as a target (page 750)*
- *Using Google Cloud Storage as a target (page 757)*
- *Using Google Dataproc as a target (page 769)*
- *Using Hadoop as a target (page 776)*
- *Using Hortonworks Data Platform (HDP) as a target (page 796)*
- *Using IBM DB2 for z/OS as a target (page 813)*
- *Using Netezza Performance Server (NPS) as a target (page 954)*
- *Using Kafka as a target (page 822)*
- *Using a Log Stream target (page 845)*
- *Using MapR Streams as a target (page 847)*
- *Using Microsoft APS PDW as a target (page 862)*
- *Using Microsoft Azure ADLS as a target (page 867)*
- *Using Microsoft Azure Database for MySQL as a target (page 881)*
- *Using Microsoft Azure Database for PostgreSQL as a target (page 889)*
- *Using Microsoft Azure Event Hubs as a target (page 896)*
- *Using Microsoft Azure HDInsight as a target (page 912)*
- *Using Microsoft Azure Synapse Analytics as a target (page 921)*
- *Using Microsoft Azure SQL Database as a target (page 931)*

- *Using Microsoft SQL Server as a target (page 938)*
- *Using a MySQL-based database as a target (page 946)*
- *Using ODBC to connect to a target (page 960)*
- *Using Oracle as a target (page 966)*
- *Using Pivotal Greenplum as a target (page 975)*
- *Using a PostgreSQL-Based database as a target (page 990)*
- *Using SAP HANA as a target (page 997)*
- *Using SAP Sybase ASE as a target (page 1002)*
- *Using SAP Sybase IQ as a target (page 1008)*
- *Using SingleStore as a target (page 1013)*
- *Using Snowflake on AWS as a target (page 1020)*
- *Using Snowflake on Azure as a target (page 1030)*
- *Using Snowflake on Google as a target (page 1038)*
- *Using Teradata Database as a target (page 1046)*
- *Using Vertica as a target (page 1056)*

10.1 Using Actian Vector as a target

This section describes how to set up and use an Actian Vector database as a target in a replication task.

In this section:

- *Prerequisites (page 582)*
- *Limitations (page 583)*
- *Permissions (page 583)*
- *Actian Vector data types (page 583)*
- *Setting general connection properties (page 585)*
- *Setting advanced connection properties (page 586)*

Prerequisites

An Actian Vector database with the tables that are being used for replication must be available to the system. Qlik Replicate supports using an Actian Vector database on both Windows and Linux.

When Actian Vector is installed on a remote machine (i.e. *not* on the Replicate machine), Actian Client Runtime 3.5 for Linux/Windows needs to be installed on the Replicate machine.

For more information about the requirements for working with Qlik Replicate, see *Installing Qlik Replicate* (page 39).

The following sections describe the prerequisites necessary to prepare your environment to work with Qlik Replicate and an Actian Vector database:

- *Actian Vector Windows environment prerequisites* (page 582)
- *Actian Vector Linux client prerequisites* (page 582)

Actian Vector Windows environment prerequisites

The following must be installed:

- Actian Vector database on any computer in your network. Make sure to configure the tables that you need for replication.
- Actian Vector DBA Tools. This includes the Ingres ODBC driver 3.5 or later. You must install the DBA tools on the same computer as the Actian Vector database. For more information, see the Actian Web site.

Actian Vector Linux client prerequisites

To set up the Actian Vector client on Linux:

1. Run the Vector client RPM file using yum.
2. Update the **/etc/odbcinst.ini** file as follows:

```
[Ingres]
Driver=/opt/Actian/Vector_Client/ingres/lib/libiodbcdriver.1.so
DriverODBCVer=3.50
DriverReadOnly=N
DriverType=Ingres
Vendor=Actian Corporation
DontDlclose=1
```

```
[Ingres CL]
Driver=/opt/Actian/Vector_Client/ingres/lib/libiodbcdriver.1.so
DriverODBCVer=3.50
DriverReadOnly=N
DriverType=Ingres
Vendor=Actian Corporation
DontDlclose=1
```

3. Update the **site_arep_login.sh** file as follows:

```
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/Actian/Vector_Client/ingres/lib
export II_SYSTEM=/opt/Actian/Vector_Client
```

4. Restart the Replicate services.
5. Change the working directory to the Vector client installation directory (by default, /opt/Actian/Vector_Client/ingres/bin/). Then run the following command:

```
export II_SYSTEM=/opt/Actian/Vector_Client
```

6. Run the `./netutil` command and setup login credentials for your Actian Vector database.

Limitations

- The following DDL operations cannot be performed on the Actian Vector database using Qlik Replicate:
 - Change schema name
 - Change column type
The replication task will fail when applying DDL changes that were captured from the source but not supported on the target.
- Full LOB data types are not supported. For information on including Limited-size LOB data types in the replication, see the *Metadata* (page 1166) tab section in *Customizing tasks* (page 1082).

Permissions

The Qlik Replicate user who is working with the Actian Vector database must be registered as a user in the Actian Vector database. This is the user that is entered in the dialog box when *Setting general connection properties* (page 585). You must grant Actian Vector account access to this user before confiding the database in Qlik Replicate.

Actian Vector data types

The Actian Vector database for Qlik Replicate supports most Actian Vector data types. The following table shows the Actian Vector target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.






*Actian Vector does not support applying changes to binary data types in **Batch optimized apply** mode. For more information on **Batch optimized apply** mode, see *Change Processing Tuning* (page 1181).*

For information on how to view the data type that is mapped from the source, see the section for the source database you are using. For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).

Qlik Replicate and Actian Vector data types

Qlik ReplicateData Types	Actian Vector Data Types
BOOLEAN	VARCHAR (length)

Qlik ReplicateData Types	Action Vector Data Types
BYTES	<p>If length is => 1 and =< 16000, then: VARCHAR (Length in Bytes)</p> <p>If length is => 16001 and =< 2147483647, then: VARCHAR (32000)</p>
DATE	ANSIDATE
TIME	TIME
DATETIME	TIMESTAMP
INT1	INTEGER1
INT2	SMALLINT
INT4	INTEGER
INT8	INTEGER8
NUMERIC	DECIMAL (p,s)
REAL4	FLOAT4
REAL8	FLOAT
STRING	<p>If length is => 1 and =< 32000, then: VARCHAR (Length)</p> <p>If length is => 32001 and =< 2147483647, then: VARCHAR (32000)</p>
UINT1	SMALLINT
UINT2	INTEGER
UINT4	INTEGER8
UINT8	INTEGER8
WSTR	<p>If length is => 1 and =< 16000, then: NVARCHAR (Length)</p> <p>If length is => 16001 and =< 2147483647, then: NVARCHAR (16000)</p>
BLOB	VARCHAR (16000)

Qlik ReplicateData Types	Action Vector Data Types
NCLOB	NVARCHAR (16000) <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">  <i>The maximum LOB size in the Metadata (page 1166) tab cannot exceed 15 KB.</i> </div>
CLOB	In Batch optimized apply mode: VARCHAR (32000) <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">  <i>In Batch optimized apply mode, the maximum LOB size in the Metadata (page 1166) tab cannot exceed 31 KB.</i> </div> In Transactional apply mode: VARCHAR (16000) <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">  <i>In Transactional apply mode, the maximum LOB size in the Metadata (page 1166) tab cannot exceed 31 KB.</i> </div> For more information on Transactional apply and Batch optimized apply , see <i>Change Processing Tuning (page 1181)</i> .

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add an Action Vector target endpoint to Qlik Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Action Vector database. This is optional.
4. Select **Target** as the database **role**.
5. Select **Action Vector Target** as the database **Type**.
6. In the **Server** field, enter the hostname or IP address of the machine on which the Action Vector database is installed.



You can use the **Advanced** tab to add specific properties and create a custom connect string. In this case, you do not need to enter information in this tab. For more information on using the **Advanced** tab, see *Setting advanced connection properties* (page 586).

7. Type the Actian Vector authentication information (**User name, Password**) for the authorized user for this Actian Vector database. If you do not know this information, see your Actian Vector system manager.



- If you are using the **Advanced** tab to create a custom string, make sure to include the **User Name** property. A **Password** can also be included but is not required. See *Setting advanced connection properties* (page 586) for more information.
- This information is case sensitive.
- If you want to set custom properties for this database, see *Setting advanced connection properties* (page 586).
- To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**. If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box. To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

Setting advanced connection properties

In the **Advanced** tab, you can set the following parameters:

- **Max file size** (KB): Select or type the maximum file size. When the Actian Vector file reaches this value, the data is loaded by the vmload utility. The default value is 300000 KB.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.2 Using Amazon MSK as a target

This section describes how to set up and use Amazon MSK as a target endpoint in a replication task. In a task with an Amazon MSK target endpoint, each source record is transformed into a message which is then written (with an optional message key) to a partition in the specified topic.

In this section:

- *Transaction processing by the consumer (page 589)*
- *Prerequisites (page 590)*
- *Limitations (page 590)*
- *Amazon MSK target data types (page 591)*
- *Setting general connection properties (page 592)*
- *Setting advanced connection properties (page 597)*
- *Data messages (page 598)*

Transaction processing by the consumer

When configuring the Qlik Replicate Amazon MSK endpoint, users can configure various settings that affect where messages are published within the Amazon MSK infrastructures (topics/partitions).

During a task's CDC stage, committed changes that are detected by the Qlik Replicate source endpoint are grouped by transaction, sorted internally in chronological order, and then propagated to the target endpoint. The target endpoint can handle the changes in various ways such as applying them to the target tables or storing them in dedicated Change Tables.

Each CDC message has both a transaction ID as well as change sequence. As the change sequence is a monotonically growing number, sorting events by change sequence always achieves chronological order. Grouping the sorted events by transaction ID then results in transactions containing chronologically sorted changes.

However, as Amazon MSK is a messaging infrastructure, applying changes is not feasible while storing changes in tables is meaningless. The Replicate Amazon MSK endpoint, therefore, takes a different approach, which is to report all transactional events as messages.

How it works

Each change in the source system is translated to a data message containing the details of the change including the transaction ID and change sequence in the source. The data message also includes the changed columns before and after the change. As explained above, the order in which the Amazon MSK target writes the messages is the same as order of changes within each transaction.

Once a data message is ready to be sent to Amazon MSK, the topic and partition it should go to are determined by analyzing the [endpoint settings](#) as well as potentially [transformation settings](#). For example, the user might decide to configure the endpoint in such a way that every table is sent to a different topic and set the partition strategy to "Random", meaning that each message (within the same table) will be sent to a different partition.

Transaction consistency from a consumer perspective

If maintaining transaction consistency is important for the consumer implementation, it means that although the transaction ID exists in all data messages, the challenge is to gather the messages in a way that would facilitate identifying a whole transaction. An additional challenge is getting the transaction in the original order they were committed, which could be an even greater challenge if transactions are spread across multiple topics and partitions.

The simplest way of achieving the above goal is to direct Replicate to a specific topic and a specific partition (in the [endpoint settings](#)). This means that all data messages will end up in a single partition, thus guaranteeing ordered delivery both of transactions *and* of changes within a transaction. The consuming application could then consume messages - accumulating a transaction in some intermediate memory buffer - and when a new transaction ID is detected, mark the previous transaction as completed.

Although the simple way may work, it's not very efficient at the task level as all messages end up in the same topic and partition, not necessarily utilizing the full parallelism of the Amazon MSK cluster. This may be a non-issue if there are multiple tasks, each taking advantage of a different topic/partition. In such as scenario, the gathering of messages from those tasks may very well utilize the cluster optimally.

The more generic way where data may be spread over multiple topics and partitions means that some intermediate buffer such as memory, a table in a relational database, or even other Amazon MSK topics would need to be used to collect information about transactions. Then, the transactions would need to be rebuilt by periodically (every few minutes/hours) sorting the events collected from Replicate's Amazon MSK output by the change sequence and grouping them by transaction ID.

Prerequisites

Before you can use Amazon MSK as a target endpoint in a Replicate task, the following prerequisites must be met:

- Open TCP ports to all the brokers from the Replicate Server machine
- Set permissions that will allow Qlik Replicate to write to the target topics. One way to do this is to use the Kafka ACLs script (`kafka-acls`).
- Either create a topic named **attrep_apply_exceptions** before starting the replication task or configure the brokers with `auto.create.topics.enable=true`.
Note that if this topic does not exist, the task will always fail when it encounters a data error, regardless of the [error handling policy](#).
For a description of the **attrep_apply_exceptions** table, see *Apply exceptions (page 1298)*

Limitations

When defining a task with Amazon MSK as the target endpoint, the following limitations apply:

- The Amazon MSK target endpoint does not support unlimited LOB size. Therefore, when replicating from source tables with LOB columns, *do not* select the **Allow unlimited LOB size** option.
For more information on defining LOB settings, see *Target metadata (page 1166)*.
- **Batch optimized apply** mode is not supported. If this mode is set, the task will automatically switch to **Transactional apply** mode and issue an appropriate warning.
For more information on these modes, see *Change Processing Tuning (page 1181)*.
- Store Changes mode is not supported.
For more information on Store Changes mode, see *Adding tasks (page 110)*.
- Amazon MSK topic names cannot exceed 255 characters (249 from Kafka 0.10) and can only contain the following characters:
`a-z|A-Z|0-9|. (dot)|_(underscore)|-(minus)`
If the source table names exceed the maximum permitted length or contain unsupported characters, you need to either modify the names before starting the task or define a global transformation. For information on defining global transformations, see *Starting the Global Transformation Rules wizard (page 1107)*.
- The [Ignore ALTER](#) Apply Changes setting is not supported for changes to source data types and table renaming.
- Column names must begin with `[A-Za-z_]` (letters or an underscore) followed by `[A-Za-z0-9_]` (letters, digits, or an underscore). For example, `_Test_` is a valid column name whereas `&Test` is not.
If a source column name does not adhere to this rule, then [a transformation should be used](#) to rename the column.

- The [Drop and Create table Target Table Preparation](#) option is not supported
- The [Truncate before loading Target Table Preparation](#) option is not supported
- The DDL History and Change Data Partitioning [Control tables](#) are not supported
- DDL Changes: Dropping or renaming a table is not supported

Amazon MSK target data types

The following table shows the default mapping from Qlik Replicate data types to Amazon MSK data types.

For information on source data type mappings, see the section for the source endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.



When using the JSON message format, binary values are represented as hexadecimal digits.



When using the Avro message format, logical data types are used.

Amazon MSK data mapping

Qlik Replicate Data Types	JSON Data Types	Avro Data Types
DATE	STRING	DATE Annotates an Avro INT.
TIME	STRING	TIME-MILLIS Annotates an Avro INT.
TIMESTAMP	STRING	TIMESTAMP-MICROS Annotates an Avro LONG.
STRING	STRING	STRING
WSTRING	STRING	STRING
CLOB	STRING	STRING
NCLOB	STRING	STRING
NUMERIC	STRING	DECIMAL (p,s) Annotates an Avro BYTES.
BYTES	BYTES	BYTES
BLOB	BYTES	BYTES
REAL4	FLOAT	FLOAT
REAL8	DOUBLE	DOUBLE

Qlik Replicate Data Types	JSON Data Types	Avro Data Types
INT1	INT	INT
INT2	INT	INT
INT4	INT	INT
UINT1	INT	INT
UINT2	INT	INT
UINT4	LONG	LONG
INT8	LONG	LONG
UINT8	STRING	DECIMAL (20,0) Annotates an Avro BYTES.
BOOLEAN	BOOLEAN	BOOLEAN

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see *Setting advanced connection properties* (page 597).

To define the general connection properties:

1. Click the **Manage Endpoint Connections** toolbar button.
The **Manage Endpoints Connections** dialog box opens.
2. Click the **New Endpoint Connection** toolbar button.
The **Name**, **Description**, **Type** and **Role** fields are displayed on the right.
3. In the **Name** field, specify a display name for the endpoint.
4. In the **Description** field, optionally type a description for the Amazon MSK endpoint.
5. Select **Target** as the endpoint **Role**.
6. Select **Amazon MSK** as the endpoint **Type**.
The dialog box is divided into **General** and **Advanced** tabs.
7. In the **Broker servers** field, specify one or more broker servers using the following format (for high availability):

```
server1[:port1] [, server2[:port2]]
```

Example:

```
192.168.1.100:9092,192.168.1.101:9093
```

Replicate will connect to the first available host. If a host is specified without a port then port 9092 will be used as the default.



When using SSL authentication, you must specify the broker FQDN (i.e. not the IP address).



All of the broker servers in your cluster need to be accessible to Replicate. However, you do not need to specify all of the servers in the **Broker servers** field. This is because Replicate only need to connect to one of the servers in order to retrieve the connection details for the other servers in the cluster. It is therefore best practice to specify the servers that are most likely to be available when the task is run. The servers to which Replicate produces messages is determined by the topic and partitioning topic and partitioning settings described below.

8. Continue as described below.

Security



- The public key file and private key file must all be in PEM format.
- All of the broker servers in the cluster must be configured to accept connection requests using the selected **Authentication** method.

In the **Security** section, the following options can be set:

- **Use SSL** (supports TLS 1.2): Select this option to encrypt the communication between the Replicate machine and the broker server(s). If the brokers are configured to require SSL, then you *must* select this option.
- **Authentication**: Select one of the following:
 - **None** - No authentication.
 - **Certificate** - If you select this option, you also need to provide the following information:
Note The public and private key files must be in PEM format.
 - **Public key file** - The full path to the public key file on the Replicate Server machine.
 - **Private key file** - The full path to the private key file on the Replicate Server machine.
 - **Private key password** - The password for the private key file.
 - **Username and Password (SASL/SCRAM-SHA-512)** - You can select this option to authenticate yourself using a user name and password (SASL/SCRAM-SHA-512).
Note that selecting this option also requires each broker's `server.properties` file to be configured with the corresponding SASL/SCRAM mechanism.

Message Properties

In the **Message Properties** section, set the following properties:

1. Choose **JSON** or **Avro** as the message format.
2. From the **Compression** drop-down list, optionally select one of the available compression methods (**Snappy** or **gzip**). The default is **None**.

Data Message Publishing

In the **Data Message Publishing** section, set the following properties:

1. In the **Publish the data to** field, choose one of the following:
 - **Specific topic** - to publish the data to a single topic. Either type a topic name or use the browse button to select the desired topic.
 - **Specific topic for each table** - to publish the data to multiple topics corresponding to the source table names.
The target topic name consists of the source schema name and the source table name, separated by a period (e.g. "dbo.Employees"). The format of the target topic name is important as you will need to prepare these topics in advance.



If the topics do not exist, configure the brokers with `auto.create.topics.enable=true` to enable Replicate to create the topics during runtime. Otherwise, the task will fail.

2. From the **Partition strategy** drop-down list, field, select either **Random** or **By message key**. If you select **Random**, each message will be written to a randomly selected partition. If you select **By message key**, messages will be written to partitions based on the selected **By message key** (described below).
3. From the **Message key** drop-down list, field, select one of the following:
 - **None** - To create messages without a message key.
 - **Schema and table name** - For each message, the message key will contain a combination of schema and table name (e.g. "dbo+Employees").
When **By message key** is selected as the **Partition strategy**, messages consisting of the same schema and table name will be written to the same partition.
 - **Primary key columns** - For each message, the message key will contain the value of the primary key column.
When **By message key** is selected as the **Partition strategy**, messages consisting of the same primary key value will be written to the same partition.

Metadata Message Publishing

In the **Metadata Message Publishing** section, specify whether or where to publish the message metadata.

From the **Publish** drop-down list, select one of the following options:

- **Do not publish metadata messages**
When this option is selected, only the data messages will be published.
- **Publish data schemas to the Confluent Schema Registry**
If you select this option, you must also configure the **Schema Registry Connection Properties** described below.



- *The Confluent Schema Registry option supports Avro message format only.*

Schema Registry Connection Properties

- **Schema Registry servers:** Specify one or more Schema Registry servers using the following format (for high availability):
`server1:port1[,server2[:port2]]`

Example:

192.168.1.100:8081,192.168.1.101:8081

Replicate will connect to the first available host.

- **Use SSL** (supports TLS 1.0, 1.1 and 1.2): Select this option to encrypt the data between the Replicate machine and the Schema Registry server(s). If the servers are configured to require SSL, then you *must* select this option.
 - **CA path:** Specify one of the following:
 - The full path (i.e. including the file name) to a specific CA certificate in PEM format
 - The directory containing certificate files with hash names
 - **Authentication** - Select one of the following Schema Registry authentication options:
 - **None** - No authentication.
 - **Certificate** - Select to authenticate using a certificate.
If you select this option, you also need to provide the following information:
 - **Public key file** - The full path to the public key file on the Replicate Server machine.
 - **Private key file** - The full path to the private key file on the Replicate Server machine.
 - **Private key password** - The password for the private key file.
 - **Username and Password** - Select to authenticate with a username and password.
 - **Use proxy server** - Select to publish to the Schema Registry via a proxy server.
 - **Host name** - The host name of the proxy server.
 - **Port** - The port via which to access the proxy server.
 - **Scheme** - Select which protocol to use to access the server (**HTTP** or **HTTPS**).
 - **SSL CA Path:** The location of the CA file on the Replicate Server machine when **HTTPS** is the selected Scheme.

Schema Registry Subject Properties

Subject Name Strategy



- The first strategy (Schema and Table Name Strategy) is a proprietary Qlik strategy while the other three are standard Confluence subject name strategies.
- For strategies with "Topic" in the subject name, the following should be considered:
 - When the "add \$topic column" method is used, the subject will be created only once (as the \$topic expression might create multiple subjects).
See also steps 3 and 4 in *Overriding the default settings* (page 596).
 - The "Metadata only" [Advanced run](#) option is not supported. This is because Replicate depends on the arrival of the first record per table in order to create the subject.

Select one of the available subject name strategies.

- Schema and Table Name - The default
- Topic Name
- Record Name
- Topic and Record Name - See also: *Message Format* (page 1196).

For more information on Confluent's subject name strategies, see

<https://docs.confluent.io/platform/current/schema-registry/serdes-develop/index.html#subject-name-strategy>

Subject Compatibility Mode

Select a compatibility mode from the **Subject compatibility mode** drop-down list. A description of the selected mode will appear below the drop-down list.



- Depending on the selected Subject Name Strategy, some of the compatibility modes may not be available.
- When [publishing messages to a Schema Registry](#), the default subject compatibility mode for all newly created Control Table subjects will be *None*, regardless of the selected **Subject compatibility mode**.
Should you wish the selected **Subject compatibility mode** to apply to Control Tables as well, set the `setNonCompatibilityForControlTables` internal parameter to `false`.

Overriding the default settings

A transformation can be defined that overrides the topic, partition and ordering key settings defined in the **General** tab.



Before you can define such a transformation, you first need to add a source endpoint to the task and select the tables you want to replicate.

To define a transformation:

1. Open the task you defined.
2. If you are defining a transformation for a single table, select one of the source tables. Otherwise, skip to Step 3.
3. Define a transformation that adds one of the following columns:



The columns listed below (prefixed with a \$) instruct Replicate to route the message to the desired topic and/or partition, and will not be included in the actual message itself.

- `$topic` - To write messages to a specific topic.
- `$partition` - To write messages to a specific partition.
- `$key` - To create a custom ordering key.

For information on creating a transformation for a single table, see *Defining transformations for a single table/view* (page 1084).

For information on creating a global transformation rule, see *Starting the Global Transformation Rules wizard* (page 1107).

4. Define an expression for the new column that returns the following values:
 - For a `$topic` column, the expression should return the topic name.
 - For a `$partition` column, the expression should return the partition number. Note that an error will be returned during runtime if the partition number does not exist.
 - For a `$key` column, the expression should return the ordering key contents. Note that the expression must return a non-empty value.

For information on creating expressions, see *Using the Expression Builder* (page 1137).

Setting advanced connection properties

In the **Advanced** tab, you can define advanced properties for the Amazon MSK target endpoint:

- **Message Maximum Size**

In the **Message maximum size** field, specify the maximum size of messages that the broker(s) are configured to receive (`message.max.bytes`). Replicate will not send messages larger than the maximum size.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary


You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

Data messages


This topic describes the structure and content of the Data messages produced by the Replicate Amazon MSK endpoint.

Data message		
Field	Type	Description
headers	Structure	Information about the current record.
operation	Enum	<p>The operation type.</p> <p>Full Load (Replicate transfers the existing records from source table)</p> <p>REFRESH – insert of a record during Full Load stage.</p> <p>CDC (Replicate transfers the changes from source table)</p> <p>INSERT – insertion of new record</p> <p>UPDATE – update of existing record</p> <p>DELETE – deletion of a record</p>
changeSequence	String	<p>A monotonically increasing change sequencer that is common to all change tables of a task.</p> <p>Use this field to order the records in chronological order.</p> <p>Applicable to CDC operations.</p>
timestamp	String	<p>The original change UTC timestamp.</p> <p>Applicable to CDC operations.</p>
streamPosition	String	<p>The source CDC stream position.</p> <p>Applicable to CDC operations.</p>

10 Adding and managing target endpoints

Field	Type	Description
transactionId	String	<p>The ID of the transaction that the change record belongs to.</p> <p>Use this field to gather all changes of a specific transaction.</p> <p>Applicable to CDC operations.</p>
changeMask	String	<p>Indicates which data columns were changed in the source table.</p> <p>The change mask is a string of hexadecimal digits, representing a bitmask of data columns in little-endian order. The bit position in the change mask is based on the ordinal of the column in the metadata message of that table.</p> <p>This means that if there are 10 data columns, they occupy bits 0 to 9 in the bitmask.</p> <p>If UPDATE mask is 0B hexadecimal, which is 1011 binary – it means that the columns at ordinals 1, 2 and 4 were changed.</p> <p>The following describes the bit semantics:</p> <ul style="list-style-type: none">• For INSERT records, all the inserted columns have the associated bits set.• For DELETE records, only primary-key (or unique index) columns have the associated bits set. This allows an applier to construct a DELETE statement without having to find the primary key fields from another source.• For UPDATE records, each column with a changed value will have the associated bit set. <div data-bbox="703 1384 1390 1520" style="border: 1px solid #ccc; padding: 5px;"><p> <i>LOB columns are not included in the changeMask bit.</i></p></div>

10 Adding and managing target endpoints

Field	Type	Description
columnMask	String	<p>Indicates which data columns are present in the message. Usually, this will include all of the table columns.</p> <div style="border: 1px solid gray; padding: 5px; margin: 10px 0;"> <p> <i>When replicating from an Oracle source without full supplemental logging, some columns might not be present in the data, since they could not be replicated.</i></p> </div> <p>The column mask is a string of hexadecimal digits, representing a bitmask of data columns in little-endian order. The bit position in the column mask is based on the ordinal of the column in the metadata message for that table.</p> <p>This allows the applier to distinguish a null value that is the <i>actual</i> value of the column, from a null value that represents a column which could not be replicated from the source database.</p>
externalSchemald	String	<p>The Schema ID. This will only be displayed if the Include external Schema IDheader check box is selected.</p> <p>As the Schema ID changes whenever a DDL is performed on the source table, consumer applications can use this information to determine if the message schema has changed.</p>
transactionEventCounter	Long	<p>The sequence number of the current operation in the transaction.</p> <p>This can be used to determine the order of operations within a transaction.</p>
transactionLastEvent	Boolean	<p>"True" indicates that it is the final record in the transaction whereas "False" indicates that not all of the records have been processed.</p>
data	Structure	The data of the table record
{columns}		The column names and values in the current record.
beforeData	Structure	The data of the table record, before the change
{columns}		<p>The column names and values, before the change.</p> <p>Applicable to UPDATE operation.</p>

10.3 Using Amazon EMR as a target

The following topics describe how to use Amazon EMR as a target endpoint in a Qlik Replicate task:



Hive can be configured to use the AWS Glue Data Catalog as its metastore or its own metastore. The metastore that will be used depends on your Amazon EMR Cluster configuration and requires no special configuration from a Replicate perspective.

In this section:

- *Prerequisites (page 602)*
- *Limitations (page 602)*
- *Change data partitioning on Amazon EMR (page 603)*
- *Amazon EMR endpoint target data types (page 603)*
- *Setting general connection properties (page 604)*
- *Setting advanced connection properties (page 605)*

Prerequisites

Before you begin to work with Amazon EMR as a target in Qlik Replicate, make sure that the following prerequisites have been met:

- **General:**
 - The Amazon S3 bucket you are using must be accessible from the Qlik Replicate machine.
 - The user specified in the Amazon EMR target endpoint's [Hive access settings](#) must have access to HiveServer2.
- **ODBC Driver when Replicate Server is running on Windows:**
 - Install Amazon Hive ODBC driver 2.6.9.1009 or later on the Replicate Server machine.
- **ODBC Driver when Replicate Server is running on Linux:**
 1. Install Amazon Hive ODBC driver 2.6.9.1009 or later on the Replicate Server machine.
 2. *After* the driver is installed, edit the **amazon.hiveodbc.ini** file as follows:

```
DriverManagerEncoding=UTF-16
ODBCInstLib=libodbcinst.so
```
- **Permissions:**
 - The [Access settings](#) selected in the Amazon EMR endpoint's **Storage** settings must have write access to the specified bucket folder.
 - **Hive table permissions:** Replicate requires permissions to perform the following operations on Hive tables: CREATE, DROP, DESCRIBE, and ALTER table. ALTER table may also include RENAME table, ADD/RENAME column, and CREATE/DROP partitions on tables.

Limitations

The following limitations apply:

- UPDATE/DELETE DMLs are not supported during change processing. If an UPDATE/DELETE DML was captured on the source, it will be ignored on the target and a warning will be written to the log. If the **Store Changes** option is enabled in the task settings, these records will be written to the Change Table.
- Limited LOB support only.
- Writing to "special" tables such as tables with partitions, buckets, or skews is not supported.
- Dropping columns and changing column data types or the data type length is not supported and will suspend the table in all cases *except* for data types that are mapped to STRING. Changes to the data type length of a data type mapped to STRING (e.g. VARCHAR(50) to VARCHAR(100)) will simply be ignored.
- Replication of Primary Key metadata is not supported.
- The following Control Tables are not supported as they require UPDATE/DELETE operations, which are not supported:
 - **Replication Status** (requires UPDATE).
Name on target: `attrep_status`

- **Suspended Tables** (requires DELETE).

Name on target: `attrep_suspended_tables`

For more information on Control Tables, see *Control tables* (page 1169).

- Table and column names can only contain ASCII characters.
- Column names cannot contain the '?' symbol. If needed, the '?' symbol can be replaced with a valid ASCII character [using a transformation](#).

Change data partitioning on Amazon EMR

When *Change Data Partitioning* (page 1177) is enabled, the Replicate Change Tables in Hive are partitioned by the `partition_name` column. Data files are uploaded to Amazon S3 storage, according to the [maximum size and time definition](#), and then stored in a directory under the Change Table directory. Whenever the specified partition timeframe ends, a partition is created in Hive, pointing to the Amazon S3 storage.

Information about the partitions is written to the [attrep_cdc_partitions](#) Control Table.

Amazon EMR endpoint target data types

The following table shows the Amazon EMR endpoint target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).

Amazon EMR endpoint target data types

Qlik Replicate Data Types	Amazon EMR Data Types
BOOL	BOOLEAN
BYTES	STRING
TIME	STRING
DATETIME	TIMESTAMP
DATE	DATE
INT1	TINYINT
INT2	SMALLINT
INT4	INT
INT8	BIGINT
NUMERIC	DECIMAL (p,s)
REAL4	FLOAT
REAL8	DOUBLE

Qlik Replicate Data Types	Amazon EMR Data Types
STRING	VARCHAR (Length)
UINT1	SMALLINT
UINT2	INT
UINT4	BIGINT
UINT8	DECIMAL (20,0)
WSTRING	VARCHAR (Length)
BLOB	STRING
NCLOB	STRING
CLOB	STRING

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add an Amazon EMR target endpoint to Qlik Replicate:

1. In the Qlik Replicate console, click **Manage Endpoint Connections** to open the **Manage Endpoint Connections** dialog box.
For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints* (page 114).
2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the endpoint being used.
3. In the **Description** field, type a description that helps to identify the Amazon EMR endpoint. This is optional.
4. Select **Amazon EMR** as the endpoint **Type**.
5. In the **Storage** section, set the following properties:
 - **Region:**
The region where your bucket is located. It is recommended to leave the default (**Auto-Detect**) as it usually eliminates the need to select a specific region. However, due to security considerations, for some regions (for example, AWS GovCloud) you might need to explicitly select the region. If the region you require does not appear in the regions list, select **Other** and set the code using the `regioncode` internal parameter in the endpoint's **Advanced** tab.
For a list of region codes, see the **Region availability** section in: <https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Concepts.RegionsAndAvailabilityZones.html>
 - **Access options:** Choose one of the following:
 - **Key pair**
Choose this method to authenticate with your Access Key and Secret Key.
 - **IAM Roles for EC2**

10 Adding and managing target endpoints

Choose this method if the machine on which Qlik Replicate is installed is configured to authenticate itself using an IAM role.

For information on IAM roles, see:

http://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles.html

- **Access key:** If you selected **Key pair** as your access method, enter your access key for Amazon S3.
- **Secret key:** If you selected **Key pair** as your access method, enter your secret key for Amazon S3.
- In the **Target folder** field, specify the target folder in your Amazon S3 bucket.



*The **Target folder** name should contain ASCII characters only.*

6. In the **Data Encryption** section, choose one of the following:
 - None
 - Server-Side Encryption with Amazon S3-Managed Keys (SSE-S3). This is the default.
 - Server-Side Encryption with AWS KMS-Managed Keys (SSE-KMS)
This option also requires you to specify your **KMS Key ID**.
For more information on the available server-side encryption methods, see:
<http://docs.aws.amazon.com/AmazonS3/latest/dev/serv-side-encryption.html>
7. In the **Hive Access** section:
 - a. In the **Host** field, specify the public host name or IP address of the Hive server.
 - b. In the **Port** field, optionally change the default port.
 - c. From the **Authentication type** drop-down list, choose either **Username** or **Username and Password** and then enter the required information in the designated fields.
 - d. To access Hive using SSL, select **Use SSL** and then specify the full path to a CA certificate file in PEM format in the **CA path** field.
 - e. In the **Database** field, specify the name of the Hive target database.
 - f. If you configured your Amazon EMR clusters to use the AWS Glue Data Catalog, select the **Store table metadata in AWS Glue Data Catalog** option.


Setting advanced connection properties

The table below describes the settings in the **Advanced** tab.

Advanced tab settings

Setting	Description
File Format	Expand this section to specify or view the file format settings.

10 Adding and managing target endpoints

Setting	Description
Target storage format	<p>Select either Text or Sequence.</p> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> <i>When using Text as the target storage format, new lines as part of the data values are not supported (due to an Apache Hadoop limitation). Best practice is to use Sequence as the target storage format.</i></p> </div>
Field delimiter	<p>The delimiter that will be used to separate fields in the target file. The default is \001. Note that field delimiters should be ASCII characters unless they are non-printable control characters, in which case they should be in Octal format (e.g. \247).</p>
Null value	<p>The value that will be used to indicate a null value in the target file.</p> <p>Example (where @ is the null value):</p> <pre>mike,male,295678 sara,female,@</pre>
Escape character	<p>The escape character is used to escape the field delimiter character. When a field delimiter is escaped, it is interpreted as actual data, and not as a field delimiter.</p> <p>Example (where \ is the escape character and a comma is the field delimiter):</p> <pre>sunroof\,power-steering</pre> <p>When using Other SerDe: The escape character is used to escape the quote character.</p> <p>Example (where \ is the escape character and double quotes is the quote character):</p> <pre>"\"sunroof, power-steering\""</pre>
Add metadata header	<p>When the target storage format is set to Text, you can optionally add a header row to the data files. The header row can contain the source column names and/or the intermediate (i.e. Replicate) data types.</p> <p>Example of a target file with a header row when both With column names and With data types are selected:</p> <pre>Position:DECIMAL(38,0),Color:VARCHAR(10) 1,"BLUE" 2,"BROWN" 3,"RED" ...</pre>
File Attributes	<p>Expand this section to specify or view the file attributes.</p>

10 Adding and managing target endpoints

Setting	Description
Maximum file size	Specify the maximum file size of each target file. When the data reaches the maximum size, the file will be closed and written to the specified target folder.
Compress files using	Select the compression method to use on Amazon S3 storage.
Change Processing	Expand this section to specify or view change processing settings.
Apply/store changes when File size reaches	Specify the minimum size of the data required to create a file in idle state.
Apply/store changes when Elapsed time reaches	Specify the maximum time to wait before applying the changes in idle state.
Proxy Server	Expand this section to specify or view proxy settings.
Use proxy server	Select this option to access Amazon S3 via a proxy server.
Host name	The host name of the proxy server.
Port	The port via which to access the proxy server.
User name	The user name for accessing the proxy server.
Password	The password for accessing the proxy server.
Scheme	Select which protocol to use to access the server (HTTP or HTTPS). In order to use HTTPS, you must first install the CA certificate that signed the proxy's certificate on the Replicate Server machine. Do this as follows: <ul style="list-style-type: none">• On Windows: Add the CA certificate to the Trusted Root Certification Authorities store of Local Computer• On Linux: Add the CA certificate to /etc/pki/tls/certs/ca-bundle.crt



To facilitate rapid delivery of DDL messages, files are uploaded immediately, regardless of the specified **File size reaches** or **Elapsed time reaches** values.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.4 Using Amazon Kinesis data streams as a target

This section describes how to set up and use Amazon Kinesis Data Streams as a target endpoint in a replication task. In a task with a Amazon Kinesis Data Streams target endpoint, each source record is transformed into a message which is then written (with an optional message key) to a shard in the specified stream.

In this section:

- *Prerequisites (page 610)*
- *Transaction processing by the consumer (page 611)*
- *Limitations (page 612)*
- *Supported data types (page 613)*
- *Setting general connection properties (page 615)*
- *Setting advanced connection properties (page 618)*
- *The Replicate envelope (page 619)*
- *Metadata and data messages (page 621)*

Prerequisites

Before you can use Amazon Kinesis Data Streams as a target endpoint in a Replicate task, the following prerequisites must be met:

- Replicate connects to AWS using SSL. This requires an appropriate CA certificate to reside on the Replicate machine; otherwise, the connection will fail. The purpose of the CA certificate is to authenticate the ownership of the AWS server certificate.
On Windows, the required CA certificate is always present whereas on Linux it may sometimes be missing. Therefore, if you are using Replicate for Linux, make sure that the required CA certificate exists in the following location on the Linux machine:
`/etc/pki/tls/certs/ca-bundle.crt`
If it does not exist, the simplest solution is to copy the certificates bundle from another Linux machine.
- The target streams must already exist before starting the replication task.
- Create a stream named **attrep_apply_exceptions** before starting the replication task. Note that if this stream does not exist, the task will always fail when it encounters a data error, regardless of the [error handling policy](#).
For a description of the **attrep_apply_exceptions** table, see *Apply exceptions (page 1298)*
- The AWS account [specified in the General tab](#) must have the following permissions:



All strings that begin with *YOUR* should be replaced with the actual value.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": [
        "kinesis:PutRecords",
        "kinesis:DescribeStream"
      ],
      "Resource": "arn:aws:kinesis:YOUR_AWS_REGION:YOUR_ACCOUNT_
NAME:stream/YOUR_STREAM_NAME"
    },
    {
      "Sid": "VisualEditor1",
      "Effect": "Allow",
      "Action": "kinesis:ListStreams",
      "Resource": "*"
    }
  ]
}
```

Additionally, if the Kinesis Stream was configured in Amazon to encrypt the data at rest, the following additional KMS permissions should be set:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": [
        "kms:Encrypt",
        "kms:DescribeKey"
      ],
      "Resource": "arn:aws:kms:YOUR_AWS_REGION:YOUR_ACCOUNT_
NAME:key/YOUR_KEY_GUID"
    },
    {
      "Sid": "VisualEditor1",
      "Effect": "Allow",
      "Action": "kms:GenerateDataKey",
      "Resource": "*"
    }
  ]
}
```

Transaction processing by the consumer

When configuring the Qlik ReplicateKafka endpoint, users can configure various settings that affect where messages are published within the Kafka infrastructures (topics/partitions).

During a task's CDC stage, committed changes that are detected by the Qlik Replicate source endpoint are grouped by transaction, sorted internally in chronological order, and then propagated to the target endpoint. The target endpoint can handle the changes in various ways such as applying them to the target tables or storing them in dedicated Change Tables.

Each CDC message has both a transaction ID as well as change sequence. As the change sequence is a monotonically growing number, sorting events by change sequence always achieves chronological order. Grouping the sorted events by transaction ID then results in transactions containing chronologically sorted changes.

However, as Kafka is a messaging infrastructure, applying changes is not feasible while storing changes in tables is meaningless. The ReplicateKafka endpoint, therefore, takes a different approach, which is to report all transactional events as messages.

How it works

Each change in the source system is translated to a data message containing the details of the change including the transaction ID and change sequence in the source. The data message also includes the changed columns before and after the change.

Once a data message is ready to be sent to Kafka, the topic and partition it should go to are determined by analyzing the endpoint settings and any transformation settings. For example, the user might decide to configure the endpoint in such a way that every table is sent to a different topic and set the partition strategy to "Random", meaning that each message (within the same table) will be sent to a different partition.

Transaction consistency from a consumer perspective

If maintaining transaction consistency is important for the consumer implementation, it means that although the transaction ID exists in all data messages, the challenge is to gather the messages in a way that would facilitate identifying a whole transaction. An additional challenge is getting the transaction in the original order they were committed, which could be an even greater challenge if transactions are spread across multiple topics and partitions.

Although the simple way may work, it's not very efficient at the task level as all messages end up in the same topic and partition, not necessarily utilizing the full parallelism of the Kafka cluster. This may be a non-issue if there are multiple tasks, each taking advantage of a different topic/partition. In such as scenario, the gathering of messages from those tasks may very well utilize the cluster optimally.

The more generic way where data may be spread over multiple topics and partitions means that some intermediate buffer such as memory, a table in a relational database, or even other topics would need to be used to collect information about transactions. Then, the transactions would need to be rebuilt by periodically (every few minutes/hours) sorting the events collected from Replicate's Kafka output by the change sequence and grouping them by transaction ID.

Limitations

When defining a task with Amazon Kinesis Data Streams as the target endpoint, the following limitations apply:

- The Amazon Kinesis Data Streams target endpoint does not support unlimited LOB size. Therefore, when replicating from source tables with LOB columns, *do not* select the **Allow unlimited LOB size** option.
For more information on defining LOB settings, see *Target metadata (page 1166)*.
- **Batch optimized apply** mode is not supported. If this mode is set, the task will automatically switch to **Transactional apply** mode and issue an appropriate warning.
For more information on these modes, see *Change Processing Tuning (page 1181)*.
- Store Changes replication mode is not supported.
For more information on Store Changes mode, see *Adding tasks (page 110)*.
- The [Ignore ALTER](#) Apply Changes setting is not supported for changes to source data types and table renaming.
- Column names must begin with [A-Za-z_] (letters or an underscore) followed by [A-Za-z0-9_] (letters, digits, or an underscore). For example, **_Test_** is a valid column name whereas **&Test** is not.
If a source column name does not adhere to this rule, then [a transformation should be used](#) to rename the column.
- The [Drop and Create table](#) **Target Table Preparation** option is not supported
- The [Truncate before loading](#) **Target Table Preparation** option is not supported

10 Adding and managing target endpoints

- The DDL History and Change Data Partitioning [Control tables](#) are not supported
- DDL Changes: Dropping or renaming a table is not supported

Supported data types

The following table shows the default mapping from Qlik Replicate data types to Amazon Kinesis Data Streams data types.

For information on source data type mappings, see the section for the source endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).




When using the JSON message format, binary values are represented as hexadecimal digits.

Supported data types

Qlik Replicate Data Types	Amazon Kinesis Data Streams Target Data Types in Schema Messages
DATE	DATE
TIME	TIME
DATETIME	DATETIME
BYTES	BYTES (length)
BLOB	BLOB
REAL4	REAL4 (7)
REAL8	REAL8 (14)
INT1	INT1 (3)
INT2	INT2 (5)
INT4	INT4 (10)
INT8	INT8 (19)
UINT1	UINT1 (3)
UINT2	UINT2 (5)
UINT4	UINT4 (10)

Values larger than $2^{31}-1$ are not supported.

Qlik Replicate Data Types	Amazon Kinesis Data Streams Target Data Types in Schema Messages
UINT8 <div style="border: 1px solid gray; padding: 5px; display: inline-block;">  Values larger than $2^{63}-1$ are not supported. </div>	UINT8 (20)
NUMERIC	NUMERIC (p,s)
STRING	STRING (Length)
WSTRING	STRING (Length)
CLOB	CLOB
NCLOB	NCLOB
BOOLEAN	BOOLEAN (1)

Mapping to JSON and Avro

Avro message format uses logical types for a more precise representation of the data type.

Data type mappings

Qlik Replicate Data Types	JSON or Avro Primitive Data Types	Avro Logical Data Types
DATE	STRING	DATE Annotates an Avro INT.
TIME	STRING	TIME-MILLIS Annotates an Avro INT.
TIMESTAMP	STRING	TIMESTAMP-MICROS Annotates an Avro LONG.
STRING	STRING	-
WSTRING	STRING	-
CLOB	STRING	-
NCLOB	STRING	-
NUMERIC	STRING	DECIMAL (p,s) Annotates an Avro BYTES.
BYTES	BYTES	-
BLOB	BYTES	-
REAL4	FLOAT	-

Qlik Replicate Data Types	JSON or Avro Primitive Data Types	Avro Logical Data Types
REAL8	DOUBLE	-
INT1	INT	-
INT2	INT	-
INT4	INT	-
UINT1	INT	-
UINT2	INT	-
UINT4	LONG	-
INT8	LONG	-
UINT8	STRING	DECIMAL (20,0) Annotates an Avro BYTES.
BOOLEAN	BOOLEAN	-

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To define the general connection properties:

1. Click the **Manage Endpoint Connections** toolbar button.
The **Manage Endpoints Connections** dialog box opens.
2. Click the **New Endpoint Connection** toolbar button.
The **Name**, **Description**, **Type** and **Role** fields are displayed on the right.
3. In the **Name** field, specify a display name for the endpoint.
4. In the **Description** field, optionally type a description for the Kafka endpoint.
5. Select **Target** as the endpoint **Role**.
6. Select **Amazon Kinesis Data Streams** as the endpoint **Type**.
The dialog box is divided into **General** and **Advanced** tabs.
7. In the **Access Details** section, set the following properties:
 - **Region:** Your Amazon Kinesis Data Streams region. If your region does not appear in the regions list, select **Other** and set the code using the regioncode internal parameter in the endpoint's **Advanced** tab.

For a list of region codes, see the **Region availability** section in:

<https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Concepts.RegionsAndAvailabilityZones.html>

- **Access options:** Choose one of the following:
 - **Key pair**
Choose this method to authenticate with your Access Key and Secret Key.

- **IAM Roles for EC2**

Choose this method if the machine on which Qlik Replicate is installed is configured to authenticate itself using an IAM role.

For information on IAM roles, see:

http://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles.html

- **Access key:** If you selected **Key pair** as your access method, enter your access key for Amazon Kinesis Data Streams.
- **Secret key:** If you selected **Key pair** as your access method, enter your secret key for Amazon Kinesis Data Streams.

8. In the **Message Properties** section, select **JSON** or **Avro** as the message **Format**.



Qlik provides an Avro Message Decoder SDK for consuming Avro messages produced by Qlik Replicate. You can download the SDK as follows:

1. Go to [Product Downloads](#).
2. Select **Qlik Data Integration**.
3. Scroll down the **Product** list and select **Replicate**.
4. In the **Download Link** column, locate the **QlikReplicate_<version>_Avro_Decoder_SDK.zip** file. Before starting the download, check the **Version** column to make sure that the version correlates with the Replicate version you have installed.
5. Proceed to download the **QlikReplicate_<version>_Avro_Decoder_SDK.zip** file.

For usage instructions, see [Kafka Avro consumers API](#) (page 1339).

An understanding of the Qlik envelope schema is a prerequisite for consuming Avro messages produced by Qlik Replicate. If you do not wish to use the SDK, see [The Qlik Envelope](#) for a description of the Qlik envelope schema.

9. In the **Data Message Publishing** section, set the following properties:

- a. In the **Publish the data to** field, choose one of the following:

- **Specific stream** - to publish the data to a single stream. Either type a stream name or use the browse button to select the desired stream.
- **Separate stream for each table** - to publish the data to multiple streams corresponding to the source table names.
The target stream name consists of the source schema name and the source table name, separated by a period (e.g. "dbo.Employees"). The format of the target stream name is important as you will need to prepare these streams in advance.

- b. From the **Partition strategy** drop-down list, field, select either **Random** or **By Partition Key**. If you select **Random**, each message will be written to a randomly selected partition. If you select **By Partition Key**, messages will be written to partitions based on the selected **Partition key** (described below).

- c. From the **Partition key** drop-down list, field, select one of the following:



The partition key is represented as a string, regardless of the selected data message format (JSON/Avro).

- **Schema and table name** - For each message, the partition key will contain a combination of schema and table name (e.g. "dbo+Employees"). Messages consisting of the same schema and table name will be written to the same partition.
- **Primary key columns** - For each message, the partition key will contain the value of the primary key column. Messages consisting of the same primary key value will be written to the same partition.

10. In the **Metadata Message Publishing** section, specify whether or where to publish the message metadata.

From the **Publish** drop-down list, select one of the following options:

- **Do not publish metadata messages.**
- **Publish metadata messages to a dedicated metadata stream**
If you select this option, either type the **Specific stream** name or use the Browse button to select the desired stream.



It is strongly recommended not to publish metadata messages to the same stream as data messages.

Overriding the default settings

A transformation can be defined that overrides the topic and message key settings defined in the **General** tab.



Before you can define such a transformation, you first need to add a source endpoint to the task and select the tables you want to replicate.

To define a transformation:

1. Open the task you defined.
2. If you are defining a transformation for a single table, select one of the source tables. Otherwise, skip to Step 3.
3. Define a transformation that adds one of the following columns:



The columns listed below (prefixed with a \$) instruct Replicate to route the message to the desired stream and will not be included in the actual message itself.

- \$stream - To write messages to a specific stream.
- \$key - To create a custom message key.

For information on creating a transformation for a single table, see *Defining transformations for a single table/view (page 1084)*.

For information on creating a global transformation rule, see *Starting the Global Transformation Rules wizard (page 1107)*.

4. Define an expression for the new column that returns the following values:
 - For a `$stream` column, the expression should return the topic name.
 - For a `$key` column, the expression should return the message key contents. Note that the expression must return a non-empty value.

For information on creating expressions, see *Using the Expression Builder (page 1137)*.

Setting advanced connection properties

In the **Advanced** tab, you can set the following advanced properties for the Amazon Kinesis Data Streams target endpoint:

Connection properties

Option	Description
Message Maximum Size	In the Message maximum size field, specify the maximum size of messages that Amazon Kinesis Data Streams is configured to receive (<code>message.max.bytes</code>). Replicate will not send messages larger than the maximum size.
Use proxy server	Select this option to access Amazon Kinesis Data Streams via a proxy server.
Host name	The host name of the proxy server.
Port	The port via which to access the proxy server.
User name	The user name for accessing the proxy server.
Password	The password for accessing the proxy server.
Scheme	Select which protocol to use to access the server (HTTP or HTTPS). In order to use HTTPS, you must first install the CA certificate that signed the proxy's certificate on the Replicate Server machine. Do this as follows: <ul style="list-style-type: none">• On Windows: Add the CA certificate to the Trusted Root Certification Authorities store of Local Computer• On Linux: Add the CA certificate to <code>/etc/pki/tls/certs/ca-bundle.crt</code>

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

The Replicate envelope

All Replicate message types covered in this section are encapsulated in a single message schema called the Replicate envelope. The schema of the Replicate envelope is as following:

```
{
  "type": "record",
  "name": "MessageEnvelope",
  "fields": [
    {"name": "magic", "type": {"type": "fixed", "name": "Magic", "size": 5}},
    {"name": "type", "type": "string"},
    {"name": "headers", "type": [{"null"}, {"type": "map", "values": "string"}]},
    {"name": "messageSchemaId", "type": [{"null"}, "string"]},
    {"name": "messageSchema", "type": [{"null"}, "string"]},
    {"name": "message", "type": "bytes"}
  ]
}
```

The fields in the envelope are as follows:

- **magic** (5 bytes fixed field)
The constant "atMSG" is used to identify this form of message. The "atMSG" constant should be used to validate that this message is indeed an Replicate envelope message.
- **type** (string field)
Describes the enveloped message type. This can be one of two values: MD which stands for metadata message and DT which stands for data message.
- **headers** (map of string key and value)
A free for use map for various properties set at the application level. Currently, no headers are set by Qlik Replicate but this may change in future versions.
- **messageSchemaId** (null or string)
A reference to a schema defined elsewhere, which can be used to deserialize the bytes in the message field. This specification does not explain how the schema ID is used for looking up the actual schema - it is an application level detail. This field is used exclusively with the messageSchema field.
- **messageSchema** (null or string)
An embedded UTF-8 encoded Avro JSON schema with which the message field can be serialized. This field is used exclusively with the messageSchemaId field.
- **message** (bytes)
An Avro encoded message, which is the payload of the message envelope.

Given the envelope schema, it is possible for anyone using this schema to properly decode the envelope messages from Kafka.

Once the envelope message has been decoded, there are two possible scenarios:

- **Scenario 1:** Decoding a self-describing message such as the metadata message
- **Scenario 2:** Decoding a message by referenced schema ID such as data messages

The method for logically decoding messages in both scenarios is described below.

Decoding a self-describing message

When the `messageschema` field is not null, it means the `message` field can be decoded using the schema included in the `messageschema` field. This is fairly straightforward to perform programatically since the only thing you need to usually supply Avro is a schema and a message, both of which are provided in the envelope message.

The Replicate metadata messages which include both table metadata, lineage and data schema description (to be referenced later by data messages) are enveloped in the self-describing envelope.

Decoding a message by referenced schema ID

Avro schemas are JSON documents which can be quite large, usually much larger than the data encoded by Avro conforming to the schema. For example, a schema of a 10 column table could be a JSON document of more than 100 characters while an actual row encoding of 10 columns may be only 10 bytes (depending of course on the type and length of fields). It is therefore typically not recommended to include schema and data together in a Kafka message because the schema information is redundant and is the same for all data messages while the actual data is the only thing which differs between data messages.

To avoid sending schema with each data message, each schema has a 32 bytes long ID. When a data message based on a previously sent data message schema (via the metadata message) is constructed, the `messageschema` field is set to null and the `messageschemaId` field is set to the 32 bytes ID of the schema instead. The application responsibility is to locate the data schema sent earlier in the metadata message and use that schema to decode the data message contained in the `message` field.

Typical consumer logic

A typical scenario involving Kafka involves Qlik Replicate as the Producer of messages into Kafka and customer code as the Consumer. Qlik Replicate offers the ability to define a specific topic as the schema topic and different topics for the table data.

The customer's consumer code should read metadata messages from the schema topic and then save the data schemas and any other information the consumer wishes to access later in a customer defined zone. Another set of customer consumers should read data messages from the various data topics, and access the data schemas zone as required to retrieve the data schemas required for decoding the data messages.

When consuming data messages and metadata messages from several topics and partitions in a multi-thread/process manner, a situation may arise where a given consumer may attempt to read a data message *before* the corresponding metadata message has been read. As it is not possible to read a data message before its corresponding metadata message, the consumer's logic should wait a reasonable amount of time until the corresponding metadata message has been read. If the metadata message is still not available after

waiting for a reasonable amount of time, the consumer should handle this as an unexpected error and activate the planned error policy. An example of such a policy could be saving the message in a dedicated “delayed” topic for later processing.

As a rule of thumb, the number of metadata messages will be much lower (in the magnitude of 1:10000 or more) than the number of data messages. So, assuming a metadata consumer is active, the gap between metadata message and data message should be no more than a few seconds (usually, milliseconds).

Metadata and data messages

This topic describes the structure and content of the metadata and data messages produced by the Amazon Kinesis Data Streams target endpoint.

Metadata message

Metadata message

Field	Type	Description
schemald	String	The unique identifier of the Avro schema.
lineage	Structure	Information about the origin of the data (Replicate server, task, table, and so on)
server	String	The name of the Replicate server.
task	String	The name of the task.
schema (Schema Name)	String	The name of the database schema.
table (Table Name)	String	The name of the table.
tableVersion	Integer	Replicate maintains a version number of the structure of source table. Upon DDL change on the source, the version is increased and a new metadata message is produced.
timestamp	String	The date and time of the metadata message.
tableStructure	Structure	Describes the structure of the table.
tableColumns	Structure	Contains the list of columns and their properties.
{columns}	Structure	For each column, a record with the below properties.
ordinal	Integer	The position of the column in the record.
type	String	The column data type.
length	Integer	The maximum size of the data (in bytes) permitted for the column.
precision	Integer	For NUMERIC data type, the maximum number of digits required to represent the value.
scale	Integer	For NUMERIC data type, the maximum number of digits to the right of the decimal point permitted for a number.

10 Adding and managing target endpoints


Field	Type	Description
primaryKeyPosition	Integer	The position of the column in the table's Primary Key. or Unique Index. The value is zero if the column is not part of the table's Primary Key.
dataSchema	String	The Avro schema for deserializing the Data messages.

Data message


Data message

Field	Type	Description
schema (name)	String	The name of the source database schema containing the replicated source table(s).
table (name)	String	The name of the source table.
headers	Structure	Information about the current record.
operation (Operation)	Enum	The operation type. Full Load - Initial load of the source data to the target topic(s) REFRESH – Insertion of a record to the target during Full Load CDC - Applies source table changes to the relevant topic INSERT – Insertion of new target record UPDATE – Update of an existing target record DELETE – Deletion of a target record
changeSequence (Change Sequence)	String	A monotonically increasing change sequencer that is common to all change tables of a task. Use this field to order the records in chronological order. Applicable to CDC operations only.
timestamp (Timestamp)	String	The original change UTC timestamp. Applicable to CDC operations only.
streamPosition	String	The source CDC stream position. Applicable to CDC operations only.
transactionId (Transaction ID)	String	The ID of the transaction that the change record belongs to. Use this field to gather all changes of a specific transaction. Applicable to CDC operations only.

10 Adding and managing target endpoints

Field	Type	Description
changeMask (Change Mask)	String	<p>Indicates which data columns were changed in the source table.</p> <p>The change mask is a string of hexadecimal digits, representing a bitmask of data columns in little-endian order. The bit position in the change mask is based on the ordinal of the column in the metadata message of that table.</p> <p>This means that if there are 10 data columns, they occupy bits 0 to 9 in the bitmask.</p> <p>If UPDATE mask is 0B hexadecimal, which is 1011 binary – it means that the columns at ordinals 1, 2 and 4 were changed.</p> <p>The following describes the bit semantics:</p> <ul style="list-style-type: none">• For INSERT records, all the inserted columns have the associated bits set.• For DELETE records, only primary-key (or unique index) columns have the associated bits set. This allows an applier to construct a DELETE statement without having to find the primary key fields from another source.• For UPDATE records, each column with a changed value will have the associated bit set. <div data-bbox="715 1227 1390 1339"> <i>LOB columns are not included in the changeMask bit.</i></div>

10 Adding and managing target endpoints

Field	Type	Description
columnMask (Column Mask)	String	<p>Indicates which data columns are present in the message. Usually, this will include all of the table columns.</p> <div style="border: 1px solid gray; padding: 5px; margin: 10px 0;">  <i>When replicating from an Oracle source without full supplemental logging, some columns might not be present in the data, since they could not be replicated.</i> </div> <p>The column mask is a string of hexadecimal digits, representing a bitmask of data columns in little-endian order. The bit position in the column mask is based on the ordinal of the column in the metadata message for that table.</p> <p>This allows the applier to distinguish a null value that is the <i>actual</i> value of the column, from a null value that represents a column which could not be replicated from the source database.</p>
externalSchemald	String	<p>The Schema ID. This will only be displayed if the Include external Schema ID header check box is selected.</p> <p>As the Schema ID changes whenever a DDL is performed on the source table, consumer applications can use this information to determine if the message schema has changed.</p>
transactionEventCounter (Transaction Event Counter)	Long	<p>The sequence number of the current operation in the transaction.</p> <p>This can be used to determine the order of operations within a transaction.</p>
transactionLastEvent (Transaction Last Event)	Boolean	<p>"True" indicates that it is the final record in the transaction whereas "False" indicates that not all of the records have been processed.</p>
data	Structure	The data of the table record
{columns}		The column names and values in the current record.
beforeData	Structure	The data of the table record, before the change
{columns}		<p>The column names and values, before the change.</p> <p>Applicable to UPDATE operation.</p>

10.5 Using Amazon Redshift as a target

This section describes how to set up and use Amazon Redshift as a target in a replication task. Amazon Redshift is located in the cloud and is accessed through an Amazon Web Services (AWS) account.

In this section:

- *Introducing the Amazon Redshift target endpoint for Qlik Replicate (page 626)*
- *Limitations (page 626)*
- *Prerequisites (page 626)*
- *Supported data types (page 628)*
- *Setting general connection parameters (page 630)*
- *Setting advanced connection properties (page 632)*

Introducing the Amazon Redshift target endpoint for Qlik Replicate

Amazon Redshift is a fully-managed petabyte-scale data warehouse service in the cloud.

In the first stage of the replication process, Qlik Replicate moves the data files created by the source database into an Amazon S3 bucket. The files are then loaded into the proper tables in the Amazon Redshift data warehouse (using the "copy" command).

The Amazon Redshift database provides full automation for:

- Schema generation and data type mapping
- Full load of source database tables
- Incremental load of changes made to source tables
- Application of DDL changes made to the source tables.
- Synchronization between full load and CDC processes.

Manual control is also available if needed.

Limitations

When using the Amazon Target endpoint, the following limitation(s) apply:

- The `ALTER TABLE <NAME> MODIFY COLUMN <NAME> <DATA_TYPE> DDL` is not supported.
- The [Allow unlimited LOB size](#) option in the **Target Metadata** task settings is not supported.
- If you need to replicate 4-byte emoji characters, use a transformation to convert the data type from `WSTRING(n)` to `WSTRING (n*2)`.

Prerequisites

The following topic describe the prerequisites necessary for working with the Amazon Redshift database.

Get started with Amazon Redshift

Once you register for an Amazon Web Services (AWS) account, you can launch an Amazon Redshift cluster and download the required SQL client tools. The following describes what you need to do to get started using Amazon Redshift as a Qlik Replicate target database.

- Sign up for an Amazon Web Services account. Then use the AWS Management Console to launch an Amazon Redshift cluster. You should note the basic information about your AWS account and your Amazon Redshift cluster, such as your password and user name. You will need this information to configure Qlik Replicate to work with the Amazon Redshift database. For more information, see *Setting general connection parameters (page 630)*.
- Make sure the time and time zone settings on the Replicate machine are accurate. This is required in order to ensure:
 - Proper synchronization of Full Load and CDC tasks.
 - Correlation of the transaction log time with the actual time.

10 Adding and managing target endpoints

- Download and install the Windows or Linux SQL client tools (according to your Replicate Server platform) necessary to connect to the Amazon Redshift cluster. Qlik Replicate requires you to download and install Amazon Redshift (x64) version 1.4.34.1000 or later.

On Linux, after installing the ODBC driver, edit the `/etc/odbcinst.ini` file as follows:

```
[ODBC Drivers]
Amazon Redshift (x64)=Installed
and
[Amazon Redshift (x64)]
Description=Amazon Redshift ODBC Driver (64-bit)
Driver=/opt/amazon/redshiftdbc/lib/64/libamazonredshiftdbc64.so
```

For a list of drivers supported by Amazon Redshift, see

<http://docs.aws.amazon.com/redshift/latest/mgmt/configure-odbc-connection.html>.

By default, Qlik Replicate uses the Amazon Redshift (x64) driver. If you use a different driver, you must change this in the Amazon Redshift database settings in the Qlik Replicate Console. For more information, see *Setting advanced connection properties* (page 632).



The driver name in the **Drivers** tab of the **ODBC Data Source Administrator x64** utility must be identical to the driver name specified in the **Advanced** tab of the endpoint settings. Otherwise, the following error will occur: "Data source name not found and no default driver specified".



To avoid conflicts when installing the driver on a Linux machine, Qlik Replicate must be installed before you install the driver. Install the Amazon Redshift ODBC driver with `--force` in the command, as in the following example:

```
rpm -ivh AmazonRedshiftODBC-64bit-1.4.34.1000-1.x86_64.rpm --force
```

Once the driver is installed, edit the **amazon.redshiftdbc.ini** file as follows:

```
DriverManagerEncoding=UTF-16
ODBCInstLib=libodbcinst.so
```

For information on signing up for an Amazon Web Services account, launching an Amazon Redshift cluster, and installing the client tools, see the Amazon Redshift Getting Started page at <http://docs.aws.amazon.com>.

Sign up for an Amazon S3 Bucket

You need to have an Amazon S3 bucket, preferably (for best performance) located in your Amazon Redshift cluster region. You must be able to access your Amazon S3 bucket directly from the machine.

For information on signing up for Amazon S3, see <http://aws.amazon.com/s3/>.

Permissions

Amazon S3 Bucket

The following permissions are needed on your Amazon S3 bucket:

- Read
- Write
- Delete

Amazon Redshift

Grant the following permissions on the Redshift database:

- Grant COPY
- Grant INSERT
- Grant UPDATE
- Grant DELETE
- Grant SELECT
- Create Schema (only required if schemas do not exist and you want Replicate to create them)
- Grant CREATE TABLE
- Grant ALTER TABLE
- Grant DROP TABLE

Security

Ensure that the following security prerequisites have been met:

- Firewall port 5439 (Amazon Redshift Cluster) needs to be opened for outbound communication.
- Replicate connects to AWS using SSL. This requires an appropriate CA certificate to reside on the Replicate machine; otherwise, the connection will fail. The purpose of the CA certificate is to authenticate the ownership of the AWS server certificate.
On Windows, the required CA certificate is always present whereas on Linux it may sometimes be missing. Therefore, if you are using Replicate for Linux, make sure that the required CA certificate exists in the following location on the Linux machine:
`/etc/pki/tls/certs/ca-bundle.crt`
If it does not exist, the simplest solution is to copy the certificates bundle from another Linux machine.

Supported data types

The Amazon Redshift database for Qlik Replicate supports most Amazon Redshift data types. The following table shows the Amazon Redshift target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.




For information on how to view the data type that is mapped from the source, see the section for the source database you are using. For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).

Redshift data types

Qlik Replicate Data Types	Amazon Redshift Data Types
BOOLEAN	BOOL
BYTES	VARCHAR (Length)

10 Adding and managing target endpoints

Qlik Replicate Data Types	Amazon Redshift Data Types
DATE	DATE
TIME	VARCHAR(20)
DATETIME	If scale is => 0 and =< 6, then: TIMESTAMP (s) If scale is => 7 and =< 12, then: VARCHAR (37)
INT1	INT2
INT2	INT2
INT4	INT4
INT8	INT8
NUMERIC	If scale is => 0 and =< 37, then: NUMERIC (p,s) If scale is => 38 and =< 127, then: VARCHAR (Length)
REAL4	FLOAT4
REAL8	FLOAT8
STRING	If length is => 1 and =< 65535, then: VARCHAR (Length in Bytes) If length is => 65535 and =< 2147483647, then: VARCHAR (65535)
UINT1	INT2
UINT2	INT4
UINT4	INT8
UINT8	NUMERIC (20,0)
WSTRING	If length is => 1 and =< 65535, then: NVARCHAR (Length in Bytes) If length is => 65536 and =< 2147483647, then: NVARCHAR (65535)

Qlik Replicate Data Types	Amazon Redshift Data Types
BLOB	VARCHAR (Max LOB Size *2) <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">  <i>The maximum LOB size in the Metadata (page 1166) tab cannot exceed 31 KB.</i> </div>
NCLOB	NVARCHAR (Max LOB Size) <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">  <i>The maximum LOB size in the Metadata (page 1166) tab cannot exceed 63 KB.</i> </div>
CLOB	VARCHAR (Max LOB Size) <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">  <i>The maximum LOB size in the Metadata (page 1166) tab cannot exceed 63 KB.</i> </div>



About Amazon Redshift LOB support:

Full LOB data types are not supported. For information on including Limited-size LOB data types in the replication, see the Metadata (page 1166) tab description in Customizing tasks (page 1082).

Setting general connection parameters

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add an Amazon Redshift Target to Qlik Replicate:

1. In the Qlik Replicate Console, click Manage Endpoint Connections to open the **Manage Endpoints Connections** dialog box.
2. In the **Manage Endpoint Connections** dialog box, click **New Endpoint Connection**.
3. In the **Name** field, type a name for your Amazon Redshift data warehouse [service]. This can be any name that will help to identify your Amazon Redshift database.
4. Optionally, in the **Description** field, type a description that helps to identify the Amazon Redshift target database.
5. Select **Target** as the role.
6. Select **Amazon Redshift** as the **Type**.
7. Enter the following **Amazon Redshift target** information:
 - **Redshift cluster:** Type the name of the Amazon Redshift cluster you are using.
 - **Port:** Type the port number for Amazon Redshift.
 - **User name:** Type an Amazon Redshift user name for a registered user.

- **Password:** Type the password for the user entered in the **User name** field.
- **Database name:** Type the **database name** or select one from the list of available Amazon Redshift data warehouse [services].

The information for these properties is available from the account page for Amazon Web Services (AWS) with the Amazon Redshift cluster. If you do not have these values, refer to your AWS account or the Amazon Redshift System Administrator for your enterprise.

8. Enter the following **Amazon S3 staging** information. You may need to click the **Amazon S3 staging** header to see the information.

- **Bucket name:** Type the name of the Amazon S3 bucket where you are copying files to.
- **Bucket region:**

The region where your bucket is located. It is recommended to leave the default (**Auto-Detect**) as it usually eliminates the need to select a specific region. However, due to security considerations, for some regions (for example, AWS GovCloud) you might need to explicitly select the region. If the region you require does not appear in the regions list, select **Other** and set the code using the `regioncode` internal parameter in the endpoint's **Advanced** tab. For a list of region codes, see the **Region availability** section in: <https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Concepts.RegionsAndAvailabilityZones.html>
- **Use AWS PrivateLink:** Select this to connect to an Amazon VPC and then specify the **VPC Endpoint URL** (for example, `https://bucket.vpce-1a2b3c4d-5e6f.s3.us-east-1.vpce.amazonaws.com`).
- **Access options:** Choose one of the following:
 - **Key pair**

Choose this method to authenticate with your Access Key and Secret Key. When this option is selected, specify the following:

 - **Access key:** Type the access key information for Amazon S3.
 - **Secret key:** Type the secret key information for Amazon S3.
 - **IAM Roles for EC2:**

Choose this method if the machine on which Qlik Replicate is installed is configured to authenticate itself using an IAM role. For more information about this access option, see: http://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles.html
 - **Security Token Service (STS)**

Choose this method to authenticate using SAML 2.0 with Active Directory Federation Services. When this option is selected, specify the following:

 - **ADFS URL:** The URL to an Active Directory Federation Services page, responsible for returning a SAML claims document to be sent over to AWS.
 - **AD principal name:** The principal (user) name to use when identifying against ADFS. The format should be: `user.name@domain`
 - **AD principal password:** The principal password to use when identifying against ADFS

- **IdP ARN:** The Amazon Resource Name (ARN) of the Active Directory issuing the SAML claims document. This is required as it enables AWS to identify the signer of the SAML document and verify its signature.
- **Role ARN:** The Amazon Resource Name (ARN) of the specific role the returned credentials should be assigned.

For more information about this access option, see:

https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles_providers_saml.html

- **Folder:** The name of the Amazon S3 folder to where you want your files to be copied.

The information for these properties is available from your Amazon Web Services (AWS) account. If you do not have these values, refer to your AWS account or the Amazon Redshift System Administrator for your enterprise



- *This information is case sensitive.*
- *To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**. If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box. To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.*

Setting advanced connection properties

In the **Advanced** tab, you can set the following parameters:

- **Max file size (MB):** Select or type the maximum size of any CSV file used to transfer data to Amazon Redshift. The default value is 1024.
- **Number of threads used to upload a file:** Select the number of threads used to upload a single file. The minimum number of threads is 1. The maximum value is 64. The default value is 10.
- **ODBC driver:** The name of the default ODBC driver you are using to connect to Amazon Redshift. The default value is **Amazon Redshift (x64)**.
- **Additional ODBC connection properties:** Type any additional ODBC connection properties if required.
- **Maximum number of files to copy per batch:** If you encounter performance issues, try adjusting this number. The default value is 8. Note that increasing the number may not necessarily improve performance *and may even degrade it*. This is because increasing the number of files also increase the load on the server machine, which may degrade performance (depending on the hardware).
- **Use proxy server** - Select this option to connect via a proxy server, and then provide the proxy connection settings:
- **Proxy connection:** Choose one of the following:
 - **Use endpoint proxy setting:** Choose this option to use the proxy settings defined for this endpoint.

- **Use default proxy settings:** Choose this option to use the default proxy settings defined for all endpoints. The default settings are defined in the Replicate server settings' [Default Proxy Server](#) tab.
- **Host name:** The host name of the proxy server.
- **Port:** The port via which to access the proxy server.
- **User name:** The user name for the proxy server.
- **Password:** The password for the proxy server.
- **Scheme:** Select which protocol to use to access the proxy server (HTTP or HTTPS). In order to use HTTPS, you must first install the CA certificate that signed the proxy server's certificate on the Replicate Server machine. Do this as follows:
 - Windows**
Add the CA certificate to the Trusted Root Certification Authorities store of Local Computer
 - Linux**
Add the CA certificate to /etc/pki/tls/certs/ca-bundle.crt

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.6 Using Amazon S3 as a target

This chapter describes how to set up and use Amazon S3 as a target in a replication task.

In this section:

- *Prerequisites (page 635)*
- *Amazon S3 target overview (page 636)*
- *Limitations and considerations (page 637)*
- *Change data partitioning (page 637)*
- *Amazon S3 target data types (page 638)*
- *Setting general connection properties (page 640)*
- *Setting advanced connection properties (page 644)*
- *Generating reference files (page 647)*
- *Content-type and content-encoding properties (page 647)*

Prerequisites

Before you can use Amazon S3 as a target endpoint in a Replicate task, the following prerequisites must be met:

- Make sure the time and time zone settings on the Replicate machine are accurate. This is required in order to ensure:
 - Proper synchronization of Full Load and CDC tasks.
 - Correlation of the transaction log time with the actual time.
- Replicate connects to AWS using SSL. This requires an appropriate CA certificate to reside on the Replicate machine; otherwise, the connection will fail. The purpose of the CA certificate is to authenticate the ownership of the AWS server certificate. On Windows, the required CA certificate is always present whereas on Linux it may sometimes be missing. Therefore, if you are using Replicate for Linux, make sure that the required CA certificate exists in the following location on the Linux machine:
`/etc/pki/tls/certs/ca-bundle.crt`
If it does not exist, the simplest solution is to copy the certificates bundle from another Linux machine.
- You must have an Amazon S3 bucket that is accessible from the Replicate machine. For information on signing up for Amazon S3, see <http://aws.amazon.com/s3/>.
- **Bucket access credentials:** Make a note of the bucket name, region, access key and secret access key - you will need to provide them in the Amazon S3 endpoint settings.
- **Bucket access permissions:** The following bucket access permissions are required:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmt1497347821000",
      "Effect": "Allow",
      "Action": [
        "s3:GetBucketLocation",
        "s3:ListBucket"
      ],
      "Resource": [
        "arn:aws:s3:::YOUR_BUCKET_NAME"
      ]
    },
    {
      "Sid": "Stmt1497344984000",
      "Effect": "Allow",
      "Action": [
        "s3:PutObject",
        "s3:GetObject",
        "s3:DeleteObject"
      ]
    }
  ]
}
```

```
    ],  
    "Resource": [  
        "arn:aws:s3:::YOUR_BUCKET_NAME/target_path",  
        "arn:aws:s3:::YOUR_BUCKET_NAME/target_path/*"  
    ]  
  }  
]  
}
```

Where `YOUR_BUCKET_NAME` is the name of your bucket and `target_path` is the intended location of the target files in your bucket.



If the target path is the bucket root, just specify `"/target_path"` with an empty string.

Amazon S3 target overview

When using Amazon S3 as a target in a Replicate task, both the Full Load and Change Processing data are written to data files. Depending on the endpoint settings, the data file format can be CSV, JSON or Parquet. While the explanations in this section relate to CSV files, the same is true for JSON and Parquet files.

Full Load files are named using incremental counters e.g. `LOAD00001.csv`, `LOAD 00002.csv`, etc. whereas Apply Changes files are named using timestamps e.g. `20141029-1134010000.csv`.



When [Parallel Load](#) is used, the naming convention for Full Load files is slightly different:

`LOAD_$(SegmentID)_$(IncreasingCounter)`

Example:

```
LOAD_1_00000001 | LOAD_1_00000002 | LOAD_1_00000003 | LOAD_2_00000001  
| LOAD_2_00000002
```



*When the [Create metadata files in the target folder](#) option is enabled, a corresponding metadata file is created using the same naming format, but with a **.dfm** extension.*

For each source table, a folder is created in the specified Amazon S3 bucket. The data files are created on the Replicate Server machine and are then uploaded to the specified Amazon S3 bucket once the [File Attributes](#) (Full Load) and [Change Processing](#) upload conditions have been met.

DDL handling

When a DDL change is captured, Replicate will close the data file and also create a DFM file if the **Create metadata files in the target folder** option is enabled. When the next batch of changes arrives, Replicate will create a new data file containing the changes. Note that the DFM file created for the new data file will match the new table structure.

Limitations and considerations

The following limitations apply to the Amazon S3 target endpoint:

- The following DDLs are supported only: Truncate table, Drop table, Create table, Add Column, Rename Column, Drop Column, and Convert Data Type.
- Full LOB Mode is not supported
- UPDATE and DELETE statements are not supported in Apply Changes replication mode
- Batch optimized apply mode is not supported
- Target lookup is not supported
- The <target folder> parameter cannot include special characters
- Limitations when using Parquet file format:
 - When the [DDL Handling Policy](#) for source tables is set to **Ignore ALTER**, the RENAME COLUMN DDL (on a Primary Key column) is not supported.
 - LOB columns larger than 1 MB are not supported.
- Database names, schema names, or table names containing slash (/) or backslash (\) characters are not supported.
- The following Control Tables are not supported as they require UPDATE/DELETE operations, which are not supported:
 - **Replication Status** (requires UPDATE).
Name on target: `attrep_status`
 - **Suspended Tables** (requires DELETE).
Name on target: `attrep_suspended_tables`
For more information on Control Tables, see *Control tables (page 1169)*.

Change data partitioning

When replicating to a Amazon S3 target, for each of the source tables, a directory is created under the specified target directory. When *Change Data Partitioning (page 1177)* is enabled, an additional sub-directory is created under the corresponding table directory. The data and metadata (when the metadata option is enabled) files are located in the partition subdirectory, as in the following example:

```
{Target Directory}
  {Table_1}
    {Partition_1}
      Data files
      DFM files
    {Partition_2}
      Data files
      DFM files
    {Partition_3}
      Data files
      DFM files
  {Table_2}
    {Partition_1}
      Data files
      DFM files
    {Partition_2}
```

```

Data files
DFM files
{Partition_3}
Data files
DFM files
    
```

Information about the partitions is written to the **attrep_cdc_partitions** Control Table. For information about this table, see *Change data partitions (page 1301)*.

Amazon S3 target data types

The following table shows the default mapping from Qlik Replicate data types to Amazon S3 target data types. Note that the data type mapping is only relevant if the [Create metadata files in the target folder](#) option is enabled.

For information on source data type mappings, see the section for the source endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

Data types

Qlik Replicate Data Types	Amazon S3 Target Data Types
DATE	DATE
TIME	TIME
DATETIME	DATETIME
BYTES	BYTES (length)
BLOB	BLOB
REAL4	REAL4 (7)
REAL8	REAL8 (14)
INT1	INT1 (3)
INT2	INT2 (5)
INT4	INT4 (10)
INT8	INT8 (19)
UINT1	UINT1 (3)
UINT2	UINT2 (5)
UINT4	UINT4 (10)
UINT8	UINT8 (20)
NUMERIC	NUMERIC (p,s)
STRING	STRING (Length)
WSTRING	STRING (Length)
CLOB	CLOB

10 Adding and managing target endpoints

Qlik Replicate Data Types	Amazon S3 Target Data Types
NCLOB	NCLOB
BOOLEAN	BOOLEAN (1)

Mapping from Qlik Replicate data types to Parquet

When Parquet is set as the message format, due to the limited number of data types supported by Parquet, the data type mappings will be as shown in the table below.

Parquet data type mappings

Replicate Data Type	Parquet Primitive Type	Logical Type
BOOLEAN	BOOLEAN	
INT1	INT32	INT(8, true)
INT2	INT32	INT(16, true)
INT4	INT32	
INT8	INT64	
UINT1	INT32	INT(8, false)
UINT2	INT32	INT(16, false)
UINT4	INT64	
UINT8	INT64	INT(64, false)
REAL4	FLOAT	
REAL8	DOUBLE	
NUMERIC	FIXED_LEN_BYTE_ARRAY (16)	DECIMAL (precision, scale)
STRING	BYTE_ARRAY	STRING
WSTRING	BYTE_ARRAY	STRING
BYTES	BYTE_ARRAY	
BLOB	BYTE_ARRAY	
CLOB	BYTE_ARRAY	STRING
NCLOB	BYTE_ARRAY	STRING
DATE	INT32	DATE
TIME	INT32	TIME(UTC=true, unit=MILLIS)
DATETIME	INT64	TIMESTAMP (UTC=true, unit=MICROS)

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add an Amazon S3 target endpoint to Qlik Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints* (page 114).
2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the endpoint being used.
3. Optionally, in the **Description** field, type a description that helps to identify the endpoint.
4. Select **Target** as the endpoint **role**.
5. Select **Amazon S3** as the endpoint **Type**.
6. Configure the remaining settings in the **General** tab as described below.

Amazon S3 Storage

This section describes the Amazon S3 storage options.

Bucket name

The name of your Amazon S3 bucket.

Bucket region

The region where your bucket is located. It is recommended to leave the default (**Auto-Detect**) as it usually eliminates the need to select a specific region. However, due to security considerations, for some regions (for example, AWS GovCloud) you might need to explicitly select the region. If the region you require does not appear in the regions list, select **Other** and specify the code in the **Region code** field.

For a list of region codes, see the **Region availability** section in:

<https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Concepts.RegionsAndAvailabilityZones.html>

Use AWS PrivateLink

Select this to connect to an Amazon VPC and then specify the **VPC Endpoint URL** (for example, <https://bucket.vpce-1a2b3c4d-5e6f.s3.us-east-1.vpce.amazonaws.com>).

Access options

Choose one of the following:

- **Key pair**
Choose this method to authenticate with your Access Key and Secret Key.
- **IAM Roles for EC2**
Choose this method if the machine on which Qlik Replicate is installed is configured to authenticate itself using an IAM role.
For more information about this access option, see:
http://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles.html

- **Security Token Service (STS)**

Choose this method to authenticate using SAML 2.0 with Active Directory Federation Services.

For more information about this access option, see:

https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles_providers_saml.html

Access key

The access key information for Amazon S3.



*This option is only available when **Key pair** is the access option.*

Secret key

The secret key information for Amazon S3.



*This option is only available when **Key pair** is the access option.*

ADFS URL

The URL to an Active Directory Federation Services page, responsible for returning a SAML claims document to be sent over to AWS.



*This option is only available when **Security Token Service (STS)** is the access option.*

AD principal name

The principal (user) name to use when identifying against ADFS

The format should be: `user.name@domain`



*This option is only available when **Security Token Service (STS)** is the access option.*

AD principal password

The principal password to use when identifying against ADFS

IdP ARN

The Amazon Resource Name (ARN) of the Active Directory issuing the SAML claims document. This is required as it enables AWS to identify the signer of the SAML document and verify its signature.

SAML Role ARN

The Amazon Resource Name (ARN) of the specific role the returned credentials should be assigned.

Switch role after assuming SAML role

Use this option to switch role after authentication.

For more information, see:

https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles_common-scenarios_aws-accounts.html

When this option is selected, the following information is required:

- **Role ARN**
The ARN associated with the target role.
- **Role external ID**
The value of the external ID condition in the target role's trust policy.

Target folder

The target folder in your Amazon S3 bucket.

File Attributes

Delimiters can be standard characters or a hexadecimal (hex) value. Note that the "0x" prefix must be used to denote a hexadecimal delimiter (e.g. 0x01 = SOH). In the **Field delimiter**, **Record delimiter** and **Null value** fields, the delimiter can consist of concatenated hex values (e.g. 0x0102 = SOHSTX), whereas in the **Quote character** and **Escape character** fields, it can only be a single hex value. The hexadecimal number 0x00 is not supported (i.e. only 0x01-0xFF are supported).

Format

You can choose to create the target files in **CSV** or **JSON**, or **Parquet** format.

In a JSON file, each record is represented by a single line, as in the following example:

```
{ "book_id": 123, "title": "Alice in Wonderland", "price": 6.99, "is_hardcover": false }  
{ "book_id": 456, "title": "Winnie the Pooh", "price": 6.49, "is_hardcover": true }  
{ "book_id": 789, "title": "The Cat in the Hat", "price": 7.23, "is_hardcover": true }
```

See also: *Content-type and content-encoding properties (page 647)*.



Changing the format (for example, from CSV to JSON or from JSON to CSV) while the task is in a stopped state and then resuming the task, is not supported.



*If you choose **JSON** or **Parquet** format, the following fields will be hidden as they are only relevant to CSV format: **Field delimiter**, **Record delimiter**, **Null value**, **Quote character**, **Escape character**, and **Add metadata header**.*

For information about data type mappings when using Parquet format and limitations, see *Mapping from Qlik Replicate data types to Parquet (page 639)* and *Limitations and considerations (page 637)*.

Field delimiter

The delimiter that will be used to separate fields (columns) in the target files. The default is a comma.

Example using a comma as a delimiter:

```
"mike","male"
```

Record delimiter

The delimiter that will be used to separate records (rows) in the target files. The default is a newline (\n).

Example:

```
"mike","male"\n"sara","female"\n
```

Null value

The string that will be used to indicate a null value in the target files.

Example (where \n is the record delimiter and @ is the null value):

```
"mike", "male", 295678\n
"sara", "female", @\n
```

Quote character

The character that will be used at the beginning and end of a text column. The default is the double-quote character ("). When a column that contains column delimiters is enclosed in double-quotes, the column delimiter characters are interpreted as actual data, and not as column delimiters.

Example (where a @ is the quote character):

```
@mike@, @male@
```

Quote escape character

The character used to escape a quote character in the actual data. The default is the double-quote character (").

Example (where " is the quote character and \ is the escape character):

```
1955, "old, \"rare\", Chevrolet", "$1000"
```

Add metadata header

When the target storage format is set to **Text**, you can optionally add a header row to the data files. The header row can contain the source column names and/or the intermediate (i.e. Replicate) data types.

Example of a target file with a header row when both **With column names** and **With data types** are selected:

```
Position:DECIMAL(38,0),Color:VARCHAR(10)
```

```
1,"BLUE"
```

```
2,"BROWN"
```

```
3,"RED"
```

```
...
```

Maximum file size

The maximum size a file can reach before it is closed (and optionally compressed). This value applies both to data files and to Reference Files.

For information on generating reference files, see *Setting advanced connection properties (page 644)*.

Compress files using

Choose one of the compression options to compress the target files or **NONE** (the default) to leave them uncompressed. Note that the available compressions options are determined by the selected file format.

Change Processing

This section describes conditional settings in **Change Processing**.

Apply/Store changes when

- **File size reaches:** Specify the maximum size of Change Data to accumulate before uploading the file to Amazon S3.

- **Elapsed time reaches:** Elapsed time reaches x.

Metadata files

This section describes the **Metadata files** option.

Create metadata files in the target folder

When this option is selected, for each data file, a matching metadata file with a **.dfm** extension will be created under the specified target folder. The metadata files provide additional information about the task/data such as the source endpoint type, the source table name, the number of records in the data file, and so on.

For a full description of the metadata file as well as possible uses, see *Metadata file description (page 1310)*.

Data encryption

This section describes the **Data encryption** options.

Choose one of the following:

- Server-Side Encryption with Amazon S3-Managed Keys (SSE-S3). This is the default.
- Server-Side Encryption with AWS KMS-Managed Keys (SSE-KMS)
This option also requires you to specify your **KMS Key ID**.
For more information on the available server-side encryption methods, see:
<http://docs.aws.amazon.com/AmazonS3/latest/dev/serv-side-encryption.html>
- None

7. To determine if the connection information you entered is correct, click **Test Connection**. If the connection test is successful, click **Save**.



*As part of connection testing process, Replicate uploads a test file to the specified Amazon S3 **Target folder** and then deletes it once a connection has been established.*

If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

*To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.*

Setting advanced connection properties

In the **Advanced** tab, you can enable the creation of reference files and set post-processing actions. These options are described below.

Post-upload processing

Run command after upload - You can process the final target files using a custom command. The command will be run whenever a data file is created.



If the **Generate a reference file** option is selected, a row (specifying the file's location) will be added to the Reference File only after the command completes successfully.

- **Command name** - The location of the command e.g. `C:\utils\move.exe`.
- **Working directory** - The directory where you want the command to run.
- **Parameters** - Specify any parameters that need to be passed to the command during runtime. You can use the following built-in parameters:
 - `${FILENAME}` - The full path to the CSV file containing the full load or CDC data.
 - `${METADATA_FILENAME}` - The full path to the DFM file containing the metadata.For information on creating metadata files, see *Setting general connection properties (page 640)*.



- If the CSV/DFM file paths contain spaces, you must enclose these parameters with quotation marks (e.g. `"${FILENAME}"`).
- The post-processing command must return a proper exit code. For information on the available exit code options, see *Exit codes for the post-processing command (page 645)*.

Generate Reference Files - Select this option to generate a Reference File (on Replicate Server) containing the full path to the Apply Changes data files.



The reference file only points to the location of the Apply Changes files, and not the Full Load files.

Reference File(s) folder - The folder on the Replicate machine in which the Reference File will be created.

Example:

`c:\temp\`

Exit codes for the post-processing command

The post-processing command must return a proper exit code. You can either use the standard exit code values described below or set a custom exit code value as described in **Setting Post Command exit codes with an internal parameter** below.

Standard Post Command exit codes

- **0** - Success
- **1** - Recoverable error. The task will recover from the point of failure according to the settings in the *Environmental errors (page 1187)* tab.
- **2** - Table error. If a table error occurs, Replicate will handle the error according to the settings in the *Table Errors (page 1191)* tab.
- **3** (or any other value e.g. -100) - Fatal error. The task will fail and not attempt recovery.

Setting Post Command Exit Codes with an Internal Parameter

You can use internal parameters to set exit codes with custom values. This is especially useful if your application already uses the standard exit code values.

See **Standard Post Command exit codes** above for a description of the exit codes.

- `successExitCode`
- `recoverableErrorExitCode`
- `tableErrorExitCode`
- `fatalErrorExitCode`

Proxy server

Use proxy server - Select this option to access Amazon S3 via a proxy server.

- **Host name** - The host name of the proxy server.
- **Port** - The port via which to access the proxy server.
- **User name** - The user name for accessing the proxy server.
- **Password** - The password for accessing the proxy server.
- **Scheme**: Select which protocol to use to access the proxy server (HTTP or HTTPS). In order to use HTTPS, you must first install the CA certificate that signed the proxy server's certificate on the Replicate Server machine. Do this as follows:

Windows

Add the CA certificate to the Trusted Root Certification Authorities store of Local Computer

Linux

Add the CA certificate to `/etc/pki/tls/certs/ca-bundle.crt`

Internal parameters

Internal parameters are parameters that are not exposed in the UI and should only be used if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your setting by clicking the View Setting Summary link. This is useful if you need to send a summary of your setting to Qlik Support.

Generating reference files

In the **Advanced** tab of the Amazon S3 target endpoint, you can enable the **Generate a reference file** option. The Reference File contains a list of the Change File locations and is therefore only relevant if the task's Apply Changes or Store Changes options are enabled.

The format of the reference file name is as follows:

```
<amazon_s3_target_endpoint_name><counter>.csv|json
```

Example:

```
AmazonS300000001.csv
```



The counter suffix increases incrementally each time a new Reference File is generated (i.e. when the file reaches the maximum size defined [in the General tab](#)). Once a new Reference File has been generated, you can delete the old reference file(s) if required.

Whenever an Apply Changes data file is created, a new row is added to the Reference File in the following format:

```
<Source_Table_Name>,<bucket_name>/<path>/<file_name>
```

Example:

```
employees,bigdata/new/files/my.company/20170611-120144192.csv
```

Note that if the **Post-process files** option in the **Advanced** tab is also enabled, the Reference File will be generated *after* the post-processing completes.

Content-type and content-encoding properties

Content-type and content-encoding properties of files uploaded to Amazon S3

Replicate Artifact	File Format	Compression	Content-Type	Content-Encoding
Data file	CSV	None	text/csv; charset=utf-8	N/A
Data file	CSV	gzip	text/csv; charset=utf-8	gzip

10 Adding and managing target endpoints

Replicate Artifact	File Format	Compression	Content-Type	Content-Encoding
Data file	JSON (lines)	None	application/x-ndjson	N/A
Data file	JSON (lines)	gzip	application/x-ndjson	gzip
Metadata file (DFM)	JSON	None	application/json	N/A

10.7 Using Cloudera Data Platform (CDP) Private Cloud as a target

This section describes how to set up and use Cloudera Data Platform (CDP) Private Cloud as the target endpoint in a replication task.

In this section:

<i>Prerequisites</i>	650
<i>Limitations and considerations</i>	650
<i>Change data partitioning</i>	651
<i>Security requirements</i>	651
<i>Supported data types</i>	651
<i>Setting general connection properties</i>	652
<i>Setting advanced connection properties</i>	655
<i>Using Kerberos authentication on Windows</i>	657

Prerequisites

Before you begin to work with CDP Private Cloud as a target in Qlik Replicate, make sure that the following prerequisites have been met:

- **General:**

- The HDFS storage must be accessible from the Qlik Replicate machine.
- When ODBC Access is selected, the Hive Server must be accessible from the Qlik Replicate machine.

- **ODBC Access:**

Accessing Hive using ODBC requires the following to be installed on the Replicate Server machine:

- **Windows:** Install Cloudera ODBC driver for Apache Hive 2.6.9 or later.
- **Linux:** Install Cloudera ODBC driver for Apache Hive 2.6.9 or later.

After the driver is installed, edit the **odbcinst.ini** file as follows:

```
DriverManagerEncoding=UTF-16
ODBCInstLib=libodbcinst.so
```

- **SSL:**

Before you can use SSL, you first need to perform the following tasks:

- Configure each NameNode and each DataNode with an SSL certificate (issued by the same CA).
- Place the CA certificate on the Replicate Server machine. The certificate should be a base64-encoded PEM (OpenSSL) file.

- **Permissions:**

The user specified in the CDP Private Cloud target settings must have write permission for the specified HDFS target directory.

- **Task Settings:**

Before starting a task with the Full Load and/or Apply Changes replication options enabled, set the Change Processing Mode to Transactional apply as Batch optimized apply is not supported.

Limitations and considerations

When working with a CDP Private Cloud endpoint, bear in mind the following limitations and considerations:

- Table and column names can only contain ASCII characters.
- Column names cannot contain the '?' symbol. If needed, the '?' symbol can be replaced with a valid ASCII character [using a transformation](#).
- Limited LOB size support only (LOB size will be limited to the maximum STRING size available on CDP Private Cloud).
- **DDL Limitations:**
 - Changing column data types or the data type length is not supported. Changing the columns data type will suspend the table and changing the data type length will simply be ignored
 - The DROP COLUMN DDL is not supported. Dropping a column during Change Capture will set NULL values in the target column position on subsequent INSERTs or UPDATES.
- **DDL limitations and considerations when the "Apply Changes DDL Handling Policy" is set to "Ignore" (Ignore DDL changes):**

- Renaming a table is not supported and will result in the table being suspended (if the Table Error handling policy is "Suspend Table", which is the default).
- Renaming a column is not supported. Renaming a column during Change Capture will set NULL values in the target column position on subsequent INSERTs or UPDATES.
- Deleting a source column will set NULLs in the corresponding target column.
- Modifications to a source column will be ignored, but may cause unpredictable behavior (for instance, when a data type is modified).
- The creation of skews, buckets or partitions in the target tables (or their preexistence in the target tables) is not supported.
- The following Control Tables are not supported as they require UPDATE/DELETE operations, which are not supported:
 - **Replication Status** (requires UPDATE).
Name on target: `attrep_status`
 - **Suspended Tables** (requires DELETE).
Name on target: `attrep_suspended_tables`
For more information on Control Tables, see *Control tables (page 1169)*.
- The **Store task recovery data in target database** Advanced Run option is not supported
- The **Batch optimized apply** Change Processing mode is not supported.

Change data partitioning

When *Change Data Partitioning (page 1177)* is enabled, the Replicate Change Tables in Hive are partitioned by the `partition_name` column. Data files are uploaded to HDFS, according to the [maximum size and time definition](#), and then stored in a directory under the Change Table directory. Whenever the specified partition timeframe ends, a partition is created in Hive, pointing to the HDFS directory.

Information about the partitions is written to the [attrep_cdc_partitions](#) Control Table.

Security requirements

- The CDP Private Cloud NameNode (and data nodes when using WebHDFS) must be accessible from the Qlik Replicate machine and the user specified in the CDP Private Cloud target settings must have write permission for the specified HDFS target directory.
- **Hive table permissions:** Replicate requires permissions to perform the following operations on Hive tables: CREATE, DROP, DESCRIBE, and ALTER table. ALTER table may also include RENAME table, ADD/RENAME column, and CREATE/DROP partitions on tables.

Supported data types

The following table shows the default mapping from the Qlik Replicate data types to the supported CDP Private Cloud target data types.

For information on how to view the data type that is mapped from the source, see the section for the source endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

Qlik ReplicateData Types	CDP Private Cloud Data Types
BOOL	BOOLEAN
BYTES	STRING
TIME	STRING
DATETIME	TIMESTAMP
DATE	DATE
INT1	TINYINT
INT2	SMALLINT
INT4	INT
INT8	BIGINT
NUMERIC	DECIMAL (p,s)
REAL4	FLOAT
REAL8	DOUBLE
STRING	VARCHAR (Length)
UINT1	SMALLINT
UINT2	INT
UINT4	BIGINT
UINT8	DECIMAL (20,0)
WSTRING	VARCHAR (Length)
BLOB	STRING
NCLOB	STRING
CLOB	STRING

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To configure the CDP Private Cloud target endpoint in Qlik Replicate:

1. In the Qlik Replicate console, click **Manage Endpoint Connections** to open the **Manage Endpoint Connections** dialog box.
For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints* (page 114).
2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the endpoint being used.

3. In the **Description** field, type a description that helps to identify the CDP Private Cloud endpoint. This is optional.
4. Select **CDP Private Cloud** as the endpoint **Type**.
5. In the **Security** section, do the following:
 - a. To encrypt the data between the Replicate machine and HDFS, select **Use SSL**. In order to use SSL, first make sure that the SSL prerequisites described in *Prerequisites (page 650)* been met. In the **CA path** field, specify one of the following items:
 - The directory containing the CA certificate.
 - The full path to a specific CA certificate.
 - b. Select one of the following authentication types:
 - **User name** - Select to connect to the CDP Private Cloud cluster with only a user name. Then, in the **User name** field, specify the name of a user authorized to access the CDP Private Cloud cluster.
 - **User name and password** - Select to connect to the CDP Private Cloud NameNode or to the Knox Gateway (when enabled - see below) with a user name and password. Then, in the **User name** and **Password** fields, specify the required user name and password.
 - **Kerberos** - Select to authenticate against the CDP Private Cloud cluster using Kerberos. Replicate automatically detects whether Qlik Replicate Server is running on Linux or on Windows and displays the appropriate settings.



Note *In order to use Kerberos authentication on Linux, the Kerberos client (workstation) package should be installed.*

Qlik Replicate Server on Linux:

When Qlik Replicate Server is running on Linux, provide the following information:

- **Realm:** The name of the realm in which your CDP Private Cloud cluster resides. For example, if the full principal name is `john.doe@EXAMPLE.COM`, then `EXAMPLE.COM` is the realm.
- **Principal:** The user name to use for authentication. The principal must be a member of the realm entered above. For example, if the full principal name is `john.doe@EXAMPLE.COM`, then `john.doe` is the principal.
- **Keytab file:** The full path of the Keytab file. The Keytab file should contain the key of the **Principal** specified above.

Qlik Replicate Server on Windows:

When Qlik Replicate Server is running on Windows, provide the following information for accessing your Active Directory KDC:




When the Replicate KDC and the CDP Private Cloud KDC are in different domains, a relationship of trust must exist between the two domains.

- **Realm:** The name of the realm/domain in which your CDP Private Cloud cluster resides (where *realm* is the MIT term while *domain* is the Active Directory term).


- **Principal:** The user name to use for authentication. The principal must be a member of the realm/domain entered above.
- **Password:** The password for the principal entered above.

If you are unsure about any of the above, consult your IT/security administrator. For additional steps required to complete setup for Kerberos authentication, see *Using Kerberos authentication on Windows (page 657)*.

6. If you need to access the Hortonworks CDP Private Cloud distribution through a Knox Gateway, select **Use Knox Gateway**. Then provide values for the following fields:

 To be able to select this option, first select **Use SSL** and then select **Username** and **Password** from the **Authentication type** drop-down list.

- **Knox Gateway host** - The FQDN (Fully Qualified Domain Name) of the Knox Gateway host.
- **Knox port** - The port number to use to access the host. The default is "8443".
- **Knox Gateway path** - The context path for the gateway. The default is "gateway".

 The port and path values are set in the **gateway-site.xml** file. If you are unsure whether the default values have been changed, contact your IT department.


- **Cluster name** - The cluster name as configured in Knox. The default is "cdp-proxy-api".

7. In the **HDFS Access** section, enter the following information:

- In the **NameNode** field, specify the IP address or hostname of the NameNode.


 This is the Active node when High Availability is enabled (see below).

- Replicate supports replication to an HDFS High Availability cluster. In such a configuration, Replicate communicates with the Active node, but switches to the Standby node in the event of failover. To enable this feature, select the **High Availability** check box. Then, specify the FQDN (Fully Qualified Domain Name) of the Standby NameNode in the **Standby NameNode** field.
- In the **Port** field, optionally change the default port (9870).
- In the **Target Folder** field, specify where to create the data files on HDFS.

 The **Target folder** name can only contain ASCII characters.

8. In the **Hive Access** section, do the following:

- a. From the **Access Hive using** drop-down list, select one of the following options:

 When the **Use Knox Gateway** option is selected or when **No Access** is selected, the **Host** and **Port** fields described below are not relevant (and are therefore hidden).




- **ODBC** - Select this option to access Hive using an ODBC driver (the default). Then continue from the **Host** field.

- **No Access** - When this option is selected, after the data files are created on HDFS, Replicate will take no further action.
- b. In the **Host** field, specify the IP address of the Hive machine.
- c. In the **Port** field, optionally change the default port (10000).
- d. In the **Database** field, specify the name of the Hive target database.

Setting advanced connection properties

The tables below describe the settings in the **Advanced** tab.

File format

Setting	Description
Target storage format	<p>Select one of the following target storage formats: Text (the default), or Sequence.</p> <div style="border: 1px solid #ccc; padding: 10px; margin-bottom: 10px;"> <p> <i>Creating and storing the tables in text format (the default) allows data to be appended to them. This in turn reduces the number of files created on CDP Private Cloud, improves query performance, and reduces the number of Hive jobs running.</i></p> </div> <div style="border: 1px solid #ccc; padding: 10px; margin-bottom: 10px;"> <p> <i>When using Text as the target storage format, new lines as part of the data values are not supported (due to an Apache Hadoop limitation). Best practice is to use Sequence as the target storage format.</i></p> </div> <div style="border: 1px solid #ccc; padding: 10px;"> <p> <i>When Sequence format is selected, the data is loaded directly to the target and stored in an external table (in sequence format). Note also that data cannot be appended in Sequence format.</i></p> </div>
Field delimiter	<p>The delimiter that will be used to separate fields in the target file. The default is \001. Note that field delimiters should be ASCII characters unless they are non-printable control characters, in which case they should be in Octal format (e.g. \247).</p>
Null value	<p>The value that will be used to indicate a null value in the target file.</p> <p>Example (where @ is the null value):</p> <p>mike,male,295678</p> <p>sara,female,@</p>
Escape character	<p>The escape character is used to escape the field delimiter character. When a field delimiter is escaped, it is interpreted as actual data, and not as a field delimiter.</p> <p>Example (where \ is the escape character and a comma is the field delimiter):</p> <p>sunroof\,power-steering</p>

10 Adding and managing target endpoints

Setting	Description
Add metadata header	<p>When the target storage format is set to Text, you can optionally add a header row to the data files. The header row can contain the source column names and/or the intermediate (i.e. Replicate) data types.</p> <p>Example of a target file with a header row when both With column names and With data types are selected:</p> <p>Position:DECIMAL(38,0),Color:VARCHAR(10)</p> <p>1,"BLUE"</p> <p>2,"BROWN"</p> <p>3,"RED"</p> <p>...</p>

File attributes

Setting	Description
Maximum file size	Specify the maximum file size of each target file. When the data reaches the maximum size, the file will be closed and written to the specified target folder.
Compress files using	Select whether or not (the default) to compress files on HDFS.

Change processing

Setting	Description
Apply/store changes when file size reaches	Specify the minimum size of the data required to apply/store the changes.
Apply/store changes when elapsed time reaches	Specify the maximum time to wait before applying/storing the changes.



- *Changes will be applied when the **Applied Changes replication** option is enabled and/or stored when the **Store Changes replication** option is enabled.*
- *To facilitate rapid delivery of DDL messages, files are uploaded immediately, regardless of the specified **File size reaches** or **Elapsed time reaches** values.*

Preventing ODBC connection timeouts

The default query timeout value is 600 seconds, which should be sufficient for most situations. However, when loading very large tables, you may need to increase the value to prevent timeouts. This can be done using the following internal parameter:

```
executeTimeout
```

See below for instructions on setting internal parameters.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

Using Kerberos authentication on Windows

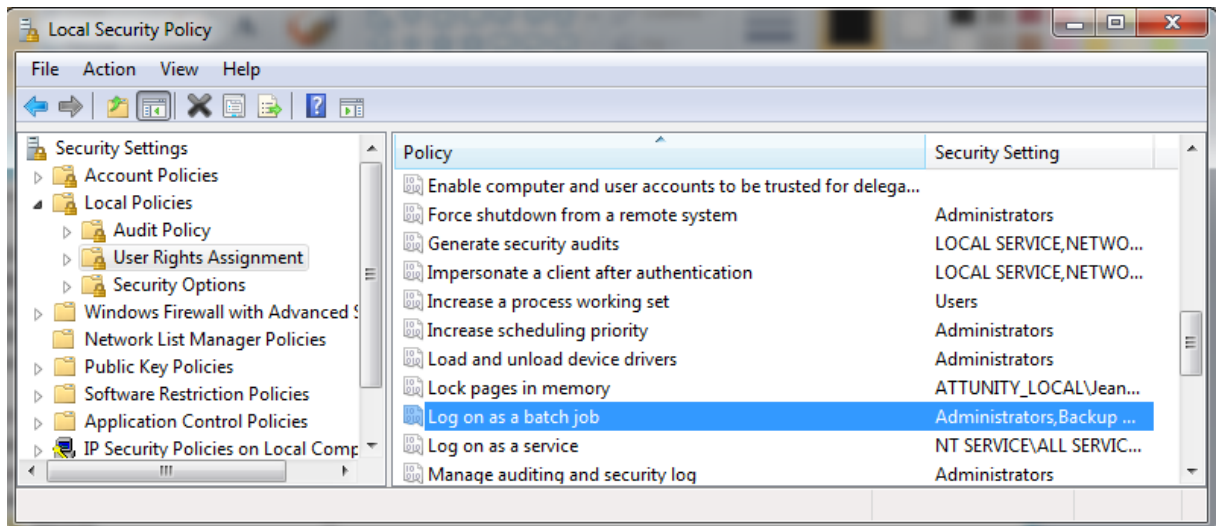
Whether Qlik Replicate Server is running on Linux or Windows, you can configure it to authenticate itself against the CDP Private Cloud cluster using Kerberos. However, when using Kerberos authentication on Windows, you need to perform the additional setup procedures outlined below.

Before beginning, make sure that the impersonated user (principal) is granted read and write permissions on the Replicate **Data** directory (<product_dir>**Data** by default) on the Qlik Replicate server. For Active Directory KDC, the impersonated user is the user configured in the user interface.

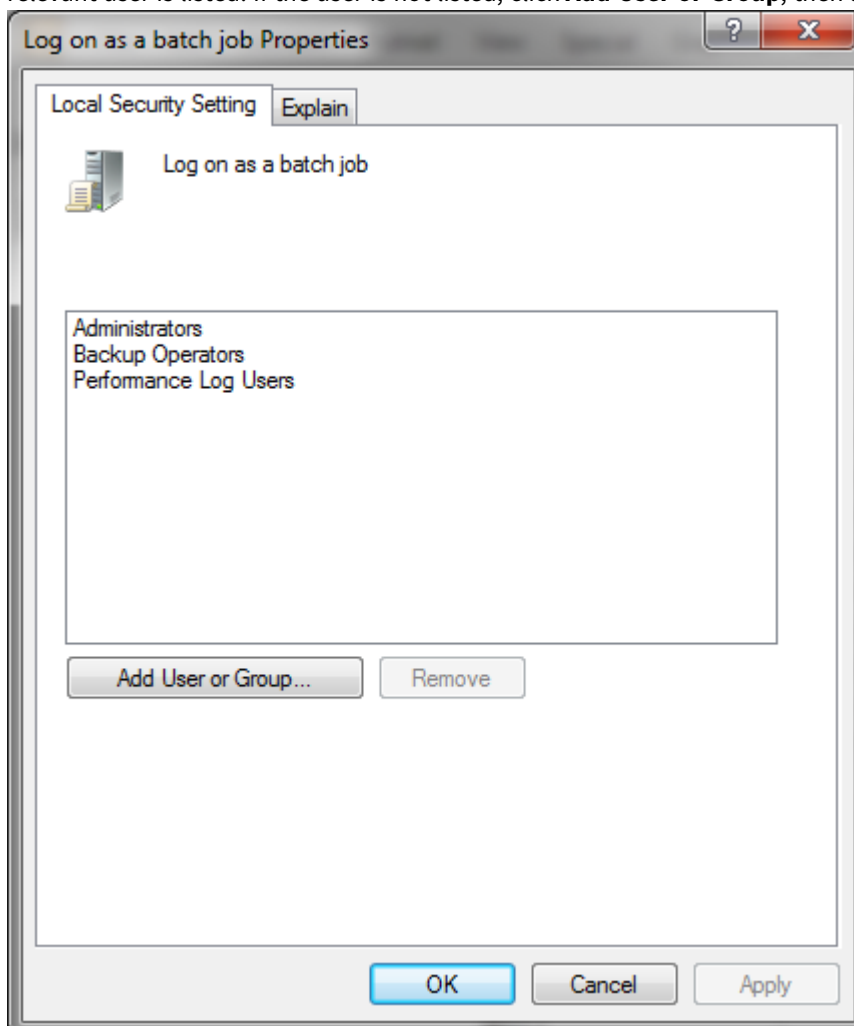
To set up Kerberos authentication on Windows, perform the following steps to ensure that the impersonated user (principal) has the **Log on as a batch job** privilege on the Qlik Replicate server.

1. On the Qlik Replicate Server machine, open the **Local Security Settings (Control Panel > System Security > Administrative Tools > Local Security Policy)**.

10 Adding and managing target endpoints



2. In the console tree, expand **Local Policies** and select **User Rights Assignments**.
3. In the details pane, double-click **Log on as a batch job**.
4. In the **Log on as a batch job Properties** dialog box, on the **Local Security Settings** tab, verify that the relevant user is listed. If the user is not listed, click **Add User or Group**, then add the user and click **OK**.



10 Adding and managing target endpoints

Your changes should take effect immediately.

10.8 Using Confluent Cloud as a target

This section describes how to set up and use Confluent Cloud as a target endpoint in a replication task.



This endpoint requires Replicate May 2023 Service Release 1 or later.

In this section:

- [Overview \(page 661\)](#)
- [Prerequisites \(page 662\)](#)
- [Limitations and considerations \(page 662\)](#)
- [Supported data types \(page 663\)](#)
- [Setting general connection properties \(page 664\)](#)
- [Setting advanced connection properties \(page 667\)](#)
- [Data messages \(page 668\)](#)

Overview

This overview summarizes the main aspects of transaction processing and transaction consistency, from a consumer perspective.

Transaction processing by the consumer

When configuring the Replicate Confluent Cloud endpoint, users can configure various settings that affect where messages are published within the Kafka infrastructures (topics/partitions).

During a task's CDC stage, committed changes that are detected by the Qlik Replicate source endpoint are grouped by transaction, sorted internally in chronological order, and then propagated to the target endpoint. The target endpoint can handle the changes in various ways such as applying them to the target tables or storing them in dedicated Change Tables.

Each CDC message has both a transaction ID as well as change sequence. As the change sequence is a monotonically growing number, sorting events by change sequence always achieves chronological order. Grouping the sorted events by transaction ID then results in transactions containing chronologically sorted changes.

However, as Kafka is a messaging infrastructure, applying changes is not feasible while storing changes in tables is meaningless. The Replicate Confluent Cloud endpoint, therefore, takes a different approach, which is to report all transactional events as messages.

How it works

Each change in the source system is translated to a data message containing the details of the change including the transaction ID and change sequence in the source. The data message also includes the changed columns before and after the change. As explained above, the order in which the Confluent Cloud endpoint writes the messages is the same as order of changes within each transaction.

Once a data message is ready to be sent to Confluent Cloud, the topic and partition it should go to are determined by analyzing the [endpoint settings](#) as well as potentially [transformation settings](#). For example, the user might decide to configure the endpoint in such a way that every table is sent to a different topic and set the partition strategy to "Random", meaning that each message (within the same table) will be sent to a different partition.

Transaction consistency from a consumer perspective

If maintaining transaction consistency is important for the consumer implementation, it means that although the transaction ID exists in all data messages, the challenge is to gather the messages in a way that would facilitate identifying a whole transaction. An additional challenge is getting the transaction in the original order they were committed, which could be an even greater challenge if transactions are spread across multiple topics and partitions.

The simplest way of achieving the above goal is to direct Replicate to a specific topic and a specific partition (in the [endpoint settings](#)). This means that all data messages will end up in a single partition, thus guaranteeing ordered delivery both of transactions *and* of changes within a transaction. The consuming application could then consume messages - accumulating a transaction in some intermediate memory buffer - and when a new transaction ID is detected, mark the previous transaction as completed.

Although the simple way may work, it's not very efficient at the task level as all messages end up in the same topic and partition, not necessarily utilizing the full parallelism of the Kafka cluster. This may be a non-issue if there are multiple tasks, each taking advantage of a different topic/partition. In such as scenario, the gathering of messages from those tasks may very well utilize the cluster optimally.

The more generic way where data may be spread over multiple topics and partitions means that some intermediate buffer such as memory, a table in a relational database, or even other Kafka topics would need to be used to collect information about transactions. Then, the transactions would need to be rebuilt by periodically (every few minutes/hours) sorting the events collected from Replicate's Kafka output by the change sequence and grouping them by transaction ID.

Prerequisites

Before you can use Confluent Cloud as a target endpoint in a Replicate task, do the following:

- Open TCP ports to the cluster the Replicate Server machine
- Set permissions that will allow Replicate to write to the target topics.
- Either create a topic named **attrep_apply_exceptions** before starting the replication task or configure the cluster with `auto.create.topics.enable=true`.



If this topic does not exist, the task will always fail when it encounters a data error, regardless of the [error handling policy](#).

For a description of the **attrep_apply_exceptions** table, see *Apply exceptions (page 1298)*

Limitations and considerations

When defining a task with Confluent Cloud as the target endpoint, the following limitations apply:

- The Confluent Cloud target endpoint does not support unlimited LOB size. Therefore, when replicating from source tables with LOB columns, *do not* select the **Allow unlimited LOB size** option. For more information on defining LOB settings, see *Target metadata (page 1166)*.
- **Batch optimized apply** mode is not supported. If this mode is set, the task will automatically switch to **Transactional apply** mode and a message will be written to the task's log file. For more information on these modes, see *Change Processing Tuning (page 1181)*.
- Store Changes mode is not supported. For more information on Store Changes mode, see *Adding tasks (page 110)*.
- The [Ignore ALTER](#) Apply Changes setting is not supported for changes to source data types and table renaming.
- Column names must begin with [A-Za-z_] (letters or an underscore) followed by [A-Za-z0-9_] (letters, digits, or an underscore). For example, **_Test_** is a valid column name whereas **&Test** is not. If a source column name does not adhere to this rule, then [a transformation should be used](#) to rename the column.
- The [Drop and Create table](#) **Target Table Preparation** option is not supported
- The [Truncate before loading](#) **Target Table Preparation** option is not supported

- The Change Data Partitioning [Control table](#) is not supported
- DDL Changes: Dropping or renaming a table is not supported

Supported data types

The table below shows the default mappings from Qlik Replicate data types to JSON and Avro logical data types.

For information on source data type mappings, see the section for the source endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

Mapping to JSON and Avro

Avro message format uses logical types for a more precise representation of the data type.

Data type mappings

Qlik Replicate Data Types	JSON	Avro
DATE	STRING	DATE Annotates an Avro INT.
TIME	STRING	TIME-MILLIS Annotates an Avro INT.
TIMESTAMP	STRING	TIMESTAMP-MICROS Annotates an Avro LONG.
STRING	STRING	STRING
WSTRING	STRING	STRING
CLOB	STRING	STRING
NCLOB	STRING	STRING
NUMERIC	STRING	DECIMAL (p,s) Annotates an Avro BYTES.
BYTES	STRING (represented as hexadecimal digits)	BYTES
BLOB	STRING (represented as hexadecimal digits)	BYTES
REAL4	NUMBER	FLOAT
REAL8	NUMBER	DOUBLE
INT1	NUMBER	INT
INT2	NUMBER	INT
INT4	NUMBER	INT

Qlik Replicate Data Types	JSON	Avro
INT8	NUMBER	LONG
UINT1	NUMBER	INT
UINT2	NUMBER	INT
UINT4	NUMBER	LONG
UINT8	STRING	DECIMAL (20,0) Annotates an Avro BYTES.
BOOLEAN	BOOLEAN	BOOLEAN

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To set the general connection properties:

1. Click the **Manage Endpoint Connections** toolbar button.
The **Manage Endpoints Connections** dialog box opens.
2. Click the **New Endpoint Connection** toolbar button.
The **Name**, **Description**, **Type** and **Role** fields are displayed on the right.
3. In the **Name** field, specify a display name for the endpoint.
4. In the **Description** field, optionally type a description for the Confluent Cloud endpoint.
5. Select **Target** as the endpoint **Role**.
6. Select **Confluent Cloud** as the endpoint **Type**.
The dialog box is divided into **General** and **Advanced** tabs.
7. Continue as described below.

Kafka access



All connections to the Kafka brokers are SSL secured.

In the **Kafka Access** section, the following options can be set:

- **Bootstrap Server:** The bootstrap server.
Example:
`pkc-159m3.us-east4.gcp.confluent.cloud:9092`
- **API Key:** The Confluent Cloud API key required to access the Kafka cluster.
- **API Secret:** The Confluent Cloud API secret required to access the Kafka cluster.

Message properties

In the **Message Properties** section, set the following properties:

- **Format:** **JSON** or **Avro**.
- **Compression:** **None** (the default), **gzip**, or **Snappy**.

Data message publishing

- In the **Publish data to** field, choose one of the following:
 - **Specific topic:** Select this option to publish the data to a single topic. Then, in the **Topic** field, either type a topic name or use the browse button to select a topic.
 - **Separate topic for each table:** Select this option to publish the data to multiple topics corresponding to the source table names.
The target topic name consists of the source schema name and the source table name, separated by a period (e.g. "dbo.Employees").
- From the **Partition strategy** drop-down list, field, select either **Random** or **By message key**. If you select **Random**, each message will be written to a randomly selected partition. If you select **By message key**, messages will be written to partitions based on the selected **By message key** (described below).
- From the **Message key** drop-down list, field, select one of the following:
 - **None** - To create messages without a message key.
 - **Schema and table name** - For each message, the message key will contain a combination of schema and table name (e.g. "dbo+Employees").
When **By message key** is selected as the **Partition strategy**, messages consisting of the same schema and table name will be written to the same partition.
 - **Primary key columns** - For each message, the message key will contain the value of the primary key column.
When **By message key** is selected as the **Partition strategy**, messages consisting of the same primary key value will be written to the same partition.

Schema Registry

In the **Schema Registry** section, you can configure the Schema Registry connection and subject properties.



- *These settings are not relevant when the message format is JSON.*
- *All connections to the Schema Registry server are SSL secured.*

Connection properties

- **Schema Registry Server:** The Schema Registry server.

Example:

```
psrc-1x2rz.us-east4.gcp.confluent.cloud:443
```

- **API Key:** The Confluent Cloud API key required to access the specified Schema Registry Server.
- **API Secret:** The Confluent Cloud API secret required to access the specified Schema Registry Server.
- **Use proxy server** - Select to publish to the Schema Registry via a proxy server.

- **Host name** - The host name of the proxy server.
- **Port** - The port via which to access the proxy server.
- **Scheme** - Select which protocol to use to access the server (**HTTP** or **HTTPS**). In order to use **HTTPS**, you must also specify the **SSL CA Path**.
- **SSL CA Path**: The location of the CA file on the Replicate Server machine when **HTTPS** is the selected Scheme.

Subject properties

- From the **Subject name strategy** drop-down list, optionally change the default subject name strategy:
 - Schema and Table Name (Default)
 - Topic Name
 - Record Name
 - Topic and Record Name - See also: *Message Format (page 1196)*.



- The default strategy (**Schema and Table Name**) is a proprietary Qlik strategy while the other three are standard Confluent subject name strategies.
- If you select **Topic Name** or **Topic and Record Name**, be aware of the following considerations:
 - When the "add \$topic column" method is used, the subject will be created only once (as the \$topic expression might create multiple subjects). See also steps 3 and 4 in *Overriding the default settings (page 666)*.
 - The "Metadata only" [Advanced run](#) option is not supported. This is because Replicate depends on the arrival of the first record per table in order to create the subject.

For more information on Confluent's subject name strategies, see

<https://docs.confluent.io/platform/current/schema-registry/serdes-develop/index.html#subject-name-strategy>

- From the **Subject compatibility mode** drop-down list, optionally change the default compatibility mode. The default is **Use Schema Registry defaults**.

For an explanation of the available modes, see <https://docs.confluent.io/platform/current/schema-registry/avro.html#compatibility-types>



When publishing messages to Confluent Schema Registry, the default subject compatibility mode for all newly created Control Table subjects will be *none*, regardless of the selected **Subject compatibility mode**. Should you wish the selected **Subject compatibility mode** to apply to Control Tables as well, set the `setNonCompatibilityForControlTables` internal parameter to `false`.

Overriding the default settings

A transformation can be defined that overrides the topic, partition and message key settings defined in the **General** tab.



Before you can define such a transformation, you first need to add a source endpoint to the task and select the tables you want to replicate.

To define a transformation:

1. Open the task you defined.
2. If you are defining a transformation for a single table, select one of the source tables. Otherwise, skip to Step 3.
3. Define a transformation that adds one of the following columns:



The columns listed below (prefixed with a *\$*) instruct Replicate to route the message to the desired topic and/or partition, and will not be included in the actual message itself.

- *\$topic* - To write messages to a specific topic.
- *\$partition* - To write messages to a specific partition.
- *\$key* - To create a custom message key.

For information on creating a transformation for a single table, see *Defining transformations for a single table/view (page 1084)*.

For information on creating a global transformation rule, see *Starting the Global Transformation Rules wizard (page 1107)*.

4. Define an expression for the new column that returns the following values:
 - For a *\$topic* column, the expression should return the topic name.
 - For a *\$partition* column, the expression should return the partition number. Note that an error will be returned during runtime if the partition number does not exist.
 - For a *\$key* column, the expression should return the message key contents. Note that the expression must return a non-empty value.

For information on creating expressions, see *Using the Expression Builder (page 1137)*.

Setting advanced connection properties

In the **Advanced** tab, you can define advanced properties for the Confluent Cloud target endpoint:

- **Message maximum size (bytes):** The maximum size of messages that the Kafka cluster is configured to receive (`message.max.bytes`). Replicate will not send messages larger than the maximum size. Adjusting the maximum message size might improve performance in certain cases.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.

10 Adding and managing target endpoints

3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary


You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

Data messages


This topic describes the structure and content of the data messages produced by the Confluent Cloud target endpoint.

Data message		
Field	Type	Description
schema (name)	String	The name of the source database schema containing the replicated source table(s).
table (name)	String	The name of the source table.
headers	Structure	Information about the current record.
operation (Operation)	Enum	The operation type. Full Load - Initial load of the source data to the target topic(s) REFRESH – Insertion of a record to the target during Full Load CDC - Applies source table changes to the relevant topic INSERT – Insertion of new target record UPDATE – Update of an existing target record DELETE – Deletion of a target record
changeSequence (Change Sequence)	String	A monotonically increasing change sequencer that is common to all change tables of a task. Use this field to order the records in chronological order. Applicable to CDC operations only.

10 Adding and managing target endpoints

Field	Type	Description
timestamp (Timestamp)	String	The original change UTC timestamp. Applicable to CDC operations only.
transactionId (Transaction ID)	String	The ID of the transaction that the change record belongs to. Use this field to gather all changes of a specific transaction. Applicable to CDC operations only.
changeMask (Change Mask)	String	<p>Indicates which data columns were changed in the source table.</p> <p>The change mask is a string of hexadecimal digits, representing a bitmask of data columns in little-endian order. The bit position in the change mask is based on the ordinal of the column in the metadata message of that table.</p> <p>This means that if there are 10 data columns, they occupy bits 0 to 9 in the bitmask.</p> <p>If UPDATE mask is 0B hexadecimal, which is 1011 binary – it means that the columns at ordinals 1, 2 and 4 were changed.</p> <p>The following describes the bit semantics:</p> <ul style="list-style-type: none">• For INSERT records, all the inserted columns have the associated bits set.• For DELETE records, only primary-key (or unique index) columns have the associated bits set. This allows an applier to construct a DELETE statement without having to find the primary key fields from another source.• For UPDATE records, each column with a changed value will have the associated bit set. <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;"> <i>LOB columns are not included in the changeMask bit.</i></div>

10 Adding and managing target endpoints

Field	Type	Description
columnMask (Column Mask)	String	<p>Indicates which data columns are present in the message. Usually, this will include all of the table columns.</p> <div style="border: 1px solid gray; padding: 5px; margin: 10px 0;"> <p> <i>When replicating from an Oracle source without full supplemental logging, some columns might not be present in the data, since they could not be replicated.</i></p> </div> <p>The column mask is a string of hexadecimal digits, representing a bitmask of data columns in little-endian order. The bit position in the column mask is based on the ordinal of the column in the metadata message for that table.</p> <p>This allows the applier to distinguish a null value that is the <i>actual</i> value of the column, from a null value that represents a column which could not be replicated from the source database.</p>
transactionEventCounter (Transaction Event Counter)	Long	<p>The sequence number of the current operation in the transaction.</p> <p>This can be used to determine the order of operations within a transaction.</p>
transactionLastEvent (Transaction Last Event)	Boolean	<p>"True" indicates that it is the final record in the transaction whereas "False" indicates that not all of the records have been processed.</p>
data	Structure	The data of the table record
{columns}		The column names and values in the current record.
beforeData	Structure	The data of the table record, before the change
{columns}		<p>The column names and values, before the change.</p> <p>Applicable to UPDATE operation.</p>

10.9 Using Databricks Lakehouse (Delta) as a target

This section describes how to set up and use Databricks Lakehouse (Delta) as the target endpoint in a replication task.



This endpoint is supported from Replicate May 2022 Service Release 02 only.

In this section:

- *Prerequisites (page 672)*
- *Limitations and considerations (page 674)*
- *Supported data types (page 675)*
- *Setting general connection properties (page 676)*
- *Setting advanced connection properties (page 679)*

Prerequisites

Before you begin to work with Databricks Lakehouse (Delta) as a target in Qlik Replicate, make sure the following prerequisites have been met.

Client prerequisites

Driver

When Replicate Server is running on Windows or Linux, download and install Simba Spark ODBC Driver 2.6.22 on the Qlik Replicate Server machine.

Replicate on Linux

When Replicate server is running on Linux, you also need to add the following section to the `/etc/odbcinst.ini` file:

```
[Simba Spark ODBC Driver]
Description=Amazon Hive ODBC Driver (64-bit)
Driver=/opt/simba/spark/lib/64/libsparkodbc_sb64.so
```

Performance and cloud services usage optimization

In order to optimize both cloud service usage and overall performance, the change processing mode must be set to Batch optimized apply in the *Change Processing Tuning (page 1181)* tab. It is also strongly recommend to enable the **Apply batched changes to multiple tables concurrently** option in the same tab.



*When the **Apply batched changes to multiple tables concurrently** option is selected, the option to set a Global Error Handling policy will not be available. Also, some of the task-specific error handling defaults will be different.*

Storage access

Databricks SQL compute must be configured to access cloud storage. For instructions, see the vendor's online help.

Permissions and access

- The time on the Qlik Replicate Server machine must be accurate.
- **Databricks table permissions:** Replicate requires permissions to perform the following operations on Databricks tables: CREATE, DROP, TRUNCATE, DESCRIBE, and ALTER table.
- In the Access Control (IAM) settings for the ADLS Gen2 file system, assign the "Storage Blob Data Contributor" role to Replicate (AD App ID). It may take a few minutes for the role to take effect.
- In order for Replicate to connect to a Databricks cluster via ODBC, users must be granted "Can Attach To" permission in their Databricks account.
- A valid security token is required to access Databricks. The token should be specified when configuring the **Databricks ODBC Access** fields in the endpoint settings.
- When configuring a new cluster with **Microsoft Azure Data Lake Storage (ADLS) Gen2**, the following line must be added to the "Spark Config" section.

```
spark.hadoop.hive.server2.enable.doAs false
```

- To be able to access the storage directories from the Databricks cluster, users need to add a configuration (in Spark Config) for that Storage Account and its key.

Example:

```
fs.azure.account.key.<storage-account-name>.dfs.core.windows.net  
<storage-account-access-key>
```

For details, refer to the Databricks online help at:

<https://docs.databricks.com/clusters/configure.html#spark-configuration>

- Best practice is *not* to use the root location (/Usr/Hive/Warehouse/) for the Databricks database as doing so may impact performance.

Staging permissions

Different permissions are required depending on the **Storage type** you select during staging, see *Setting general connection properties* (page 676).

Amazon S3

- You must have an Amazon S3 bucket that is accessible from the Replicate Server machine. For information on signing up for Amazon S3, see <http://aws.amazon.com/s3/>.
- Replicate connects to AWS using SSL. This requires an appropriate CA certificate to reside on the Replicate Server machine; otherwise, the connection will fail. The purpose of the CA certificate is to authenticate the ownership of the AWS server certificate. On Windows, the required CA certificate is always present whereas on Linux it may sometimes be missing. Therefore, if you are using Replicate for Linux, make sure that the required CA certificate exists in the following location:
`/etc/pki/tls/certs/ca-bundle.crt`
If it does not exist, the simplest solution is to copy the certificates bundle from another Linux machine.
- **Bucket access credentials:** Make a note of the bucket name, region, access key and secret access key - you will need to provide them in the Qlik Replicate Databricks Lakehouse (Delta) target settings.
- **Bucket access permissions:** Qlik Replicate requires the following bucket access permissions:

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {  
      "Sid": "Stmt1497347821000",  
      "Effect": "Allow",  
      "Action": [  
        "s3:GetBucketLocation",  
        "s3:ListBucket"  
      ],  
      "Resource": [  
        "arn:aws:s3:::YOUR_BUCKET_NAME"  
      ]  
    }  
  ]  
}
```

```
    },
    {
      "Sid": "Stmt1497344984000",
      "Effect": "Allow",
      "Action": [
        "s3:PutObject",
        "s3:GetObject",
        "s3:DeleteObject"
      ],
      "Resource": [
        "arn:aws:s3:::YOUR_BUCKET_NAME/target_path",
        "arn:aws:s3:::YOUR_BUCKET_NAME/target_path/*"
      ]
    }
  ]
}
```

Google Cloud Storage

The **JSON credentials** specified in the endpoint's **Staging** settings must be for an account that has read and write access to the specified bucket and folder.

Microsoft Azure Data Lake Storage (ADLS) Gen2

The **Application Registration Client ID** specified in the endpoint's **Staging** settings must have write access to the specified ADLS storage staging folder.

Limitations and considerations

The following limitations apply when using Databricks Lakehouse (Delta):

- Supports LOB replication in Limited LOB mode only.
- Table names can only contain alphanumeric and underscore characters.
- Proxy limitations: See *Setting advanced connection properties (page 679)*.
- General DDL limitations and considerations:
 - Dropping columns is not supported. Capturing changes to a table with a dropped column will set NULL values in the corresponding target column.
 - Renaming columns is not supported.
 - Modifying columns is not supported.
 - Renaming tables is not supported when the **Storage type** is **Amazon S3**.
- DDL limitations and considerations for the Apply Changes [DDL Handling Policy](#):
 - When set to **Ignore <DDL Type>**:
 - When renaming or modifying a column, capturing changes will set NULL values in the corresponding target column.
 - Although truncating tables is supported, capturing changes from a truncated table will cause data conflicts as the target table will not be truncated.

- Renaming tables is not supported. Capturing changes from a renamed table will result in the table being suspended (when the default table error handling policy is set). However, the table will be suspended only after the task is topped and resumed.
- When set to **<Action> target table:**
 - When renaming or modifying a column, the table will be suspended.
- Target tables with buckets or partitions will not be created automatically on the target. Preexisting Databricks Lakehouse (Delta) target tables with buckets or partitions (which are identical to those of the corresponding source tables) are supported though.
- The [Transactional apply](#) Change Processing mode is not supported.
- The Apply Changes replication mode supports tables with a Primary Key/Unique Index only. Tables without Primary Keys/Unique Indexes will be suspended at the Change Processing stage.



Possible workarounds for tables without Primary Keys/Unique Indexes:

- *Configure a transformation to add a Primary Key/Unique Index column during replication*
- *Create a [Store Changes](#) replication task for tables without Primary Keys/Unique Indexes*

- The [Store task recovery data in target database](#) Change Processing option is not supported.
- The Replication Status Control Table (Name on target: `attrep_status`) is not supported.
- If a connectivity-token to Databricks expires during a task, you must stop and resume the task once the new token is set.
- Changing the [Target table schema](#) is the equivalent of changing the specified database name in the endpoint settings.

Supported data types

The following table shows the Databricks Lakehouse (Delta) endpoint target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

Qlik Replicate and Databricks Lakehouse (Delta) data types

Qlik Replicate Data Types	Databricks Lakehouse (Delta) Data Types
BOOL	BOOLEAN
BYTES	STRING
TIME	STRING
DATETIME	TIMESTAMP
DATE	DATE

Qlik Replicate Data Types	Databricks Lakehouse (Delta) Data Types
INT1	TINYINT
INT2	SMALLINT
INT4	INT
INT8	BIGINT
NUMERIC	DECIMAL (p,s)
REAL4	FLOAT
REAL8	DOUBLE
STRING	Databricks 7.x and earlier: STRING Databricks 8.2 and later: VARCHAR (Length)
UINT1	SMALLINT
UINT2	INT
UINT4	BIGINT
UINT8	DECIMAL (20,0)
WSTRING	Databricks 7.x and earlier: STRING Databricks 8.2 and later: VARCHAR (Length)
BLOB	STRING
NCLOB	STRING
CLOB	STRING

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a Databricks Lakehouse (Delta) target endpoint to Qlik Replicate:

1. In the Qlik Replicate console, click **Manage Endpoint Connections** to open the **Manage Endpoint Connections** dialog box.
For more information on how to add an endpoint connection, see *Defining and managing endpoints (page 114)*.
2. In the **Name** field, specify a name for your endpoint.

3. In the **Description** field, provide information that helps identify the endpoint. This field is optional.
4. Select **Databricks Lakehouse (Delta)** as the endpoint **Type**.
5. In the **Databricks ODBC Access** section, provide the following information:
 - a. In the **Host** field, specify the host name of the Databricks workspace.
 - b. In the **Port** field, specify the port via which to access the workspace.
 - c. In the **Token** field, enter your personal token for accessing the workspace.
 - d. In the **HTTP Path** field, specify the path to the cluster being used.
 - e. If you want the tables to be created in Unity Catalog, select **Use Unity Catalog** and then specify the **Catalog** name.



When the **Use Unity Catalog** option is selected, you need to allow Replicate to access external (unmanaged) tables by defining an external location in Databricks. For guidelines, see:

<https://docs.databricks.com/data-governance/unity-catalog/manage-external-locations-and-credentials.html#manage-permissions-for-an-external-location>

Note that even if you select the **Managed tables** option below, you still need to define an external location. This is because the Replicate Net Changes table is always created as unmanaged (external).

- f. In the **Database** field, specify the name of the Databricks target database.
- g. Choose whether to create tables as **Managed tables** or **Unmanaged tables**. If you selected **Unmanaged tables**, specify the location for your tables in the **Location for tables** field.



When creating unmanaged tables:

- Tables are created using **CREATE** or **REPLACE**
- Tables are not dropped, they are truncated. Therefore, if a table is dropped in the source during CDC, the corresponding target table in Databricks will be truncated, not dropped.

For more information on managed versus unmanaged tables, see

<https://docs.databricks.com/lakehouse/data-objects.html>

6. In the **Staging** section, provide the information specified in *Storage types* (page 677).

Storage types

For **Microsoft Azure Data Lake Storage (ADLS) Gen2**, provide the following information:

1. In the **Storage type** field, select **Microsoft Azure Data Lake Storage (ADLS) Gen2**.
2. In the **Storage account** field, specify the name of your storage account.



To connect to an Azure resource on Government Cloud or China Cloud, you need to specify the full resource name of the storage account. For example, assuming the storage account is "myaccount", then the resource name for China Cloud would be `myaccount.dfs.core.chinacloudapi.cn`.
In addition, you also need to specify the login URL using the `adlsLoginUrl` internal parameter. For China Cloud, this would be `https://login.chinacloudapi.cn`.
For information on setting internal parameters, see *Setting advanced connection properties* (page 679).

3. In the **Azure Active Directory Tenant ID** field, specify the Azure active directory tenant ID.
4. In the **Application Registration Client ID** field, specify the application registration client ID.
5. In the **Application Registration Secret** field, specify the application registration secret.
6. In the **Container** field, specify the name of your container.
7. In the **Staging directory** field, specify where to create the data files on ADLS.



The **Staging directory** name can only contain ASCII characters.

For **Amazon S3**, provide the following information:

1. In the **Storage type** field, select **Amazon S3**.
2. In the **Bucket name** field, specify the name of your Amazon S3 bucket..
3. In the **Bucket region** field, specify the region where your bucket is located. It is recommended to leave the default (**Auto-Detect**) as it usually eliminates the need to select a specific region. However, due to security considerations, for some regions (for example, AWS GovCloud) you might need to explicitly specify the region. To do this, select **Other** and specify the code in the **Region code** field. For a list of region codes, see the **Region availability** section in: <https://docs.aws.amazon.com/general/latest/gr/s3.html>
4. In the **Access options** field, choose one of the following:
 - **Key pair**
Choose this method to authenticate with your Access Key and Secret Key.
 - **IAM Roles for EC2**
Choose this method if the machine on which Qlik Replicate is installed is configured to authenticate itself using an IAM role.
For more information about this access option, see: http://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles.html
5. In the **Access key** field, specify the access key information for Amazon S3.



This option is only available when **Key pair** is the access option.

6. In the **Secret key** field, specify the secret key information for Amazon S3.



This option is only available when **Key pair** is the access option.

7. In the **Staging directory** field, specify where to create the data files on Amazon S3.



The **Staging directory** name can only contain ASCII characters.

For **Google Cloud Storage**, provide the following information:

1. In the **Storage type** field, select **Google Cloud Storage**.
2. In the **JSON credentials** field, specify the JSON credentials for the service account key used to access the Google Cloud Storage bucket.
3. In the **Bucket name** field, specify the name of the bucket in Google Cloud Storage where you want the data files to be written. This must be the same as the bucket you configured for your Databricks cluster.
4. In the **Staging directory** field, specify where to create the data files in the specified bucket.



The **Staging directory** name can only contain ASCII characters.

Setting advanced connection properties

In the **Advanced** tab, you can set the following properties:

File attributes

Maximum file size (MB): Select or type the maximum size of any CSV file used to transfer data to Databricks Lakehouse (Delta). The default value is 100.

Proxy server



The options displayed depend on the selected storage type.

For information on the available storage types, see [Setting general connection properties \(page 676\)](#)

- **Use proxy server** - Select this option to connect via a proxy server, and then provide the proxy connection settings:
- Choose **Connect via proxy server to** options according to which resources you want to access via a proxy server:
 - **Database (via ODBC)**
 - **Staging storage**



When the **Staging storage** option is selected and the storage type is either **Google Cloud Storage** or **Microsoft Azure Data Lake Storage (ADLS) Gen2**, proxy authentication is not supported.

- **Azure Active Directory**



Only available when **Microsoft Azure Data Lake Storage (ADLS) Gen2** is the selected storage type. When **Azure Active Directory** is selected, proxy authentication is not supported.

- **Proxy connection:** Choose one of the following:
 - **Use endpoint proxy setting:** Choose this option to use the proxy settings defined for this endpoint.
 - **Use default proxy settings:** Choose this option to use the default proxy settings defined for all endpoints. The default settings are defined in the Replicate server settings' [Default Proxy Server](#) tab.
- **Host name:** The host name of the proxy server.
- **Port:** The port via which to access the proxy server.
- **User name:** The user name for the proxy server.
- **Password:** The password for the proxy server.
- **Scheme:** Select which protocol to use to access the proxy server (HTTP or HTTPS). In order to use HTTPS, you must first install the CA certificate that signed the proxy's certificate on the Replicate Server machine. Do this as follows:
 - **On Windows:** Add the CA certificate to the Trusted Root Certification Authorities store of Local Computer
 - **On Linux:** Add the CA certificate to `/etc/pki/tls/certs/ca-bundle.crt`



The **HTTPS** scheme is only supported when both the **Amazon S3** storage type and the **Staging storage** proxy option (only) are selected.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

10 Adding and managing target endpoints

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.10 Using Databricks (Cloud Storage) as a target

This section describes how to set up and use Databricks (Cloud Storage) as the target endpoint in a replication task.



Hive can be configured to use the AWS Glue Data Catalog as its metastore or its own metastore. The metastore that will be used depends on your Databricks on AWS Cluster configuration and requires no special configuration from a Replicate perspective.

In this section:

- *Prerequisites (page 683)*
- *Limitations and considerations (page 685)*
- *Change data partitioning (page 686)*
- *Supported data types (page 686)*
- *Setting general connection properties (page 688)*
- *Setting advanced connection properties (page 694)*

Prerequisites

Before you begin to work with Databricks (Cloud Storage) as a target in Qlik Replicate, make sure that the following prerequisites have been met:

General prerequisites

Required driver

When Replicate Server is running on Windows or Linux, download and install Simba Spark ODBC Driver 2.6.22 on the Qlik Replicate Server machine.

Replicate on Linux

When Replicate server is running on Linux, you also need to add the following section to the `/etc/odbcinst.ini` file:

```
[Simba Spark ODBC Driver]
Description=Amazon Hive ODBC Driver (64-bit)
Driver=/opt/simba/spark/lib/64/libsparkodbc_sb64.so
```

AWS prerequisites

Permissions

The following permissions are required:

- The "Bucket" specified in the Databricks on AWS endpoint's **Storage** settings must have write access to the specified storage target folder.
- **Databricks table permissions:** Replicate requires permissions to perform the following operations on Databricks tables: CREATE, DROP, TRUNCATE, DESCRIBE, and ALTER table. ALTER table may also include RENAME table and ADD column.
- In order for Replicate to connect to a Databricks cluster via ODBC, the user specified in the endpoint settings must be granted "Can Attach To" permission.
- The S3 storage bucket (or the directory under the bucket) must be mounted on the Databricks File System (DBFS).
For information on how to set this up, refer to <https://docs.databricks.com/data/data-sources/aws/amazon-s3.html>
- You must provide Replicate with a valid security token for access to Databricks.

Microsoft Azure prerequisites

Permissions

- The **Azure Data Lake Storage (ADLS) Gen2** file system or **Blob storage** location (whichever you are using) must be accessible from the Qlik Replicate machine.
- The "Storage account" (when using Blob storage) or "Azure Active Directory application ID" (when using ADLS) specified in the Microsoft Azure Databricks endpoint's **Storage** settings must have write access to the specified Blob/ADLS storage target folder.

- **Databricks table permissions:** Replicate requires permissions to perform the following operations on Databricks tables: CREATE, DROP, DESCRIBE, and ALTER table. ALTER table may also include RENAME table and ADD column.
- In the Access Control (IAM) settings for the ADLS Gen2 file system, assign the “Storage Blob Data Contributor” role to Replicate (AD App ID). It may take a few minutes for the role to take effect.
- In order for Replicate to connect to a Databricks cluster via JDBC/ODBC, you must have "Can Attach To" permission

General

- The Blob storage container or ADLS Data Lake Store (according to your selected storage type) must be mounted on the Databricks File System (DBFS).



- For information on how to set this up with Blob storage, see [Azure storage](#)
- For information on how to set this up with ADLS Gen2 storage, see [Azure Data Lake Gen2 storage](#)

- You must provide Replicate with a valid security token for access to Databricks.
- When configuring a new cluster with **Azure Data Lake Storage (ADLS) Gen2**, the line `"spark.hadoop.hive.server2.enable.doAs false"` must be added to the "Spark Config" section.

Supported blob storage types

The following blob storage types are supported:

- Standard storage with block blobs
- Premium storage with block blobs only

Google Cloud prerequisites

Permissions

The following permissions are required:

- The Google service account should be granted the following permissions on the bucket:
 - `storage.buckets.get`
 - `storage.objects.get`
 - `storage.objects.list`
 - `storage.objects.create`
 - `storage.objects.delete`
- The Google service account should be granted the `storage.buckets.list` permission. This permission is required in order to browse for a bucket in the endpoint settings.
- Grant the user account permission to perform the following operations on Databricks tables: CREATE, DROP, DESCRIBE, and ALTER table.

- To enable Replicate to connect to the Databricks cluster via ODBC, the user account must be granted "Can Attach To" permission.
For more information, see the [Databricks online help](#).
- You must provide Replicate with a valid security token for Databricks access.
- To access the storage directories from the Databricks cluster, you need to add a configuration for that storage account and its key. For an explanation of how to do this, see the instructions for accessing a GCS bucket directly in the [Databricks online help](#).

Limitations and considerations

The following limitations apply:

- UPDATE/DELETE DMLs are not supported during change processing. If an UPDATE/DELETE DML was captured on the source, it will be ignored on the target and a warning will be written to the log. If the **Store Changes** option is enabled in the task settings, these records will be written to the Change Table.
- Limited LOB support only.
- Unsupported DDL operations:
 - Changes to a column's data type or data type length will not be captured and will result in the table being suspended.
 - A REMOVE COLUMN operation on the source will set NULL values in the corresponding target column.
 - A RENAME COLUMN operation on the source will set NULL values in the corresponding target column.
 - The Rename Table DDL is not supported when using AWS Glue Catalog as the Hive metastore. Tables updated with Rename DDL operations will be suspended.



When Replicate is set to [ignore DDL changes](#), ADD COLUMN, RENAME TABLE, DROP TABLE, and TRUNCATE TABLE DDL operations will be ignored. Unsupported DDL operations will not be ignored, but they will also not be applied to the target. Instead, they will behave as described above.

- The following Control Tables are not supported as they require UPDATE/DELETE operations, which are not supported:
 - **Replication Status** (requires UPDATE)
Name on target: `attrep_status`
 - **Suspended Tables** (requires DELETE)
Name on target: `attrep_suspended_tables`
- Table and column names can only contain ASCII characters. Column names cannot contain the '?' symbol. If needed, the '?' symbol can be replaced with a valid ASCII character [using a transformation](#).
- Connecting via an HTTP proxy server with authentication is not supported.
- Replication to Delta tables is not supported.
- The [Batch optimized apply](#) Change Processing mode is not supported.
- The [Commit rate during full load](#) option is not relevant.

- The [Create primary key or unique index after full load completes](#) option is not relevant.
- If you change the **Databricks storage access method** endpoint setting during a task or create a new task with a Databricks target whose **Databricks storage access method** differs from that of existing tasks, you must also perform the procedures described in *Setting general connection properties* (page 688).
- As support for the VARCHAR data type was introduced in Databricks 8, creating a task that replicates to Databricks 8.x and then downgrading the cluster to an earlier Databricks version (in the same task) requires the task to be reloaded.
- Limitations when using Parquet file format:
 - When the [DDL Handling Policy](#) for source tables is set to **Ignore ALTER**, the RENAME COLUMN DDL (on a Primary Key column) is not supported.
 - LOB columns larger than 1 MB are not supported.
- Unity Catalog limitations:
 - Change Data Partitioning is not supported and should be set to “Off”. For more information, see: *Store Changes Settings* (page 1175).
 - When both **Use Unity Catalog** and **All-purpose clusters** are selected (in the endpoint settings' **General** tab), the **Sequence** storage format will not be available for selection. Choose **Text** or **Parquet** instead (in the **Advanced** tab).



If you want to enable the **Use Unity Catalog** and **All-purpose clusters** options, but already have existing target tables in Sequence format, you need to do the following:

1. Stop the task (only relevant for running tasks).
2. Select the **Target storage format** in the endpoint settings (**Text** or **Parquet**).
3. Manually migrate the target tables to be the same as the selected storage format.
4. Resume the task (only relevant for running tasks).

If you fail to do this, Replicate will suspend any tables in Sequence format, which will require you to reload the target.

Change data partitioning

When *Change Data Partitioning* (page 1177) is enabled, the Replicate Change Tables in Databricks are partitioned by the `partition_name` column. Data files are uploaded to the storage, according to the [maximum size and time definition](#), and then stored in a directory under the Change Table directory. Whenever the specified partition timeframe ends, a partition is created that points to the storage location.

Information about the partitions is written to the [attrep_cdc_partitions](#) Control Table.

Supported data types

The following table shows the supported data types and the default mapping from Qlik Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).

10 Adding and managing target endpoints

Qlik Replicate and Databricks on AWS data types

Replicate data types	Databricks data types
BOOL	BOOLEAN
BYTES	STRING
TIME	STRING
DATETIME	TIMESTAMP
DATE	DATE
INT1	TINYINT
INT2	SMALLINT
INT4	INT
INT8	BIGINT
NUMERIC	DECIMAL (p,s)
REAL4	FLOAT
REAL8	DOUBLE
STRING	Databricks 7.x and earlier: STRING Databricks 8.2 and later: VARCHAR (Length)
UINT1	SMALLINT
UINT2	INT
UINT4	BIGINT
UINT8	DECIMAL (20,0)
WSTRING	Databricks 7.x and earlier: STRING Databricks 8.2 and later: VARCHAR (Length)
BLOB	STRING
NCLOB	STRING
CLOB	STRING

Mapping from Qlik Replicate data types to Parquet

When Parquet is set as the message format, due to the limited number of data types supported by Parquet, the data type mappings will be as shown in the table below.

Parquet data type mappings

Replicate Data Type	Parquet Primitive Type	Logical Type
BOOLEAN	BOOLEAN	
INT1	INT32	INT(8, true)
INT2	INT32	INT(16, true)
INT4	INT32	
INT8	INT64	
UINT1	INT32	INT(8, false)
UINT2	INT32	INT(16, false)
UINT4	INT64	
UINT8	INT64	INT(64, false)
REAL4	FLOAT	
REAL8	DOUBLE	
NUMERIC	FIXED_LEN_BYTE_ARRAY (16)	DECIMAL (precision, scale)
STRING	BYTE_ARRAY	STRING
WSTRING	BYTE_ARRAY	STRING
BYTES	BYTE_ARRAY	
BLOB	BYTE_ARRAY	
CLOB	BYTE_ARRAY	STRING
NCLOB	BYTE_ARRAY	STRING
DATE	INT32	DATE
TIME	INT32	TIME(UTC=true, unit=MILLIS)
DATETIME	INT64	TIMESTAMP (UTC=true, unit=MICROS)

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a Databricks (Cloud Storage) target endpoint to Qlik Replicate:

1. In the Qlik Replicate console, click **Manage Endpoint Connections** to open the **Manage Endpoint Connections** dialog box.
For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints* (page 114).
2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the endpoint being used.
3. In the **Description** field, type a description that helps to identify the endpoint. This is optional.
4. Select **Databricks (Cloud Storage)** as the endpoint **Type**.

Databricks ODBC Access

Expand the **Databricks ODBC Access** section and provide the following information:

- **Host:** The host name of the Databricks workspace where the specified Amazon S3 bucket is mounted.
- **Port:** The port via which to access the workspace (you can change the default port 443, if required).
- **Token:** Your personal token for accessing the workspace.
- **HTTP Path:** The path to the cluster being used.
- If you want the tables to be created in Unity Catalog, select **Use Unity Catalog** and then specify the **Catalog** name.



When the **Use Unity Catalog** option is selected, note the following:

- **Prerequisite:** To allow Replicate to access external (unmanaged) tables, you need to define an external location in Databricks. For more information, see: <https://docs.databricks.com/data-governance/unity-catalog/manage-external-locations-and-credentials.html#manage-permissions-for-an-external-location>
- **Limitation:** Change Data Partitioning is not supported and should be set to "Off". For more information, see: *Store Changes Settings* (page 1175).

- **Database:** The name of the Databricks target database.
- **Cluster type:** Select either **All-purpose (Interactive)** or **SQL Warehouse** according to your cluster type.



When both **Use Unity Catalog** and **All-purpose clusters** are selected (in the endpoint settings' **General** tab), the **Sequence** storage format will not be available. Choose **Text** or **Parquet** instead (in the **Advanced** tab).

Cloud Storage Access

Select a **Storage type** and then configure the settings according to the sections below.

Amazon S3

- **Bucket name:** The name of your Amazon S3 bucket.
- **Bucket region:**
The region where your bucket is located. It is recommended to leave the default (**Auto-Detect**) as it usually eliminates the need to select a specific region. However, due to security considerations, for some regions (for example, AWS GovCloud) you might need to explicitly select the region. If the region you require does not appear in the regions list, select **Other** and specify the code in the **Region code** field.

For a list of region codes, see the **Region availability** section in:

<https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Concepts.RegionsAndAvailabilityZones.html>

- **Access type:** Choose one of the following:
 - **Key pair**
Choose this method to authenticate with your Access Key and Secret Key.
 - **IAM Roles for EC2**
Choose this method if the machine on which Qlik Replicate is installed is configured to authenticate itself using an IAM role.
For more information about this access option, see:
http://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles.html

- **Access key:** The access key information for Amazon S3.



*This option is only available when **Key pair** is the access option.*

- **Secret key:** The secret key information for Amazon S3.



*This option is only available when **Key pair** is the access option.*

- **Target directory:** The target folder in your Amazon S3 bucket.
- **Databricks storage access method:** Choose which method your Databricks cluster uses to access the Amazon S3 storage: **Access Directly** (the default) or **Access through DBFS Mount**. AWS S3 storage can be accessed by mounting buckets using DBFS or directly. Replicate needs to know which method Databricks uses to access the storage so that it can set the "location" property when it creates the tables in Databricks. The "location" property enables Databricks to access the storage data using its configured storage access method.
 - **Mount Path** - When **Access through DBFS Mount** is selected, you also need to specify the mount path.



The mount path cannot contain special characters or spaces.

For more information on configuring Databricks to access the Amazon S3 storage, see

<https://docs.databricks.com/data/data-sources/aws/amazon-s3.html>



All tables being replicated in a single task must be configured to access the storage using the **Databricks storage access method** defined in the endpoint settings (see earlier). The same is true for Replicate Control Tables, which are common to all tasks. Any tables that are configured to use a different storage access method will not be replicated or, in the case of Control Tables, will not be written to the target database. To prevent this from happening, you must perform the procedures below if you need to do any of the following:

- Change the Databricks storage access method during a task and retain the existing tables
- Define a new task with a Databricks target whose Databricks storage access method differs from existing tasks (with a Databricks target)

To change the Databricks storage access methods during a task:

1. Stop the task.
2. Change the **Databricks storage access method**.
3. For all tables (including Control Tables), execute the ALTER TABLE statement:
`ALTER TABLE table_identifier [partition_spec] SET LOCATION 'new_location'`
For details, see <https://docs.databricks.com/spark/latest/spark-sql/language-manual/sql-ref-syntax-ddl-alter-table.html#set-location>.
4. Start the task.

To create a new task with a Databricks endpoint whose "Databricks storage access method" differs from existing tasks:

- When you define the task, specify a dedicated Control Table schema in the task settings' Control tables (page 1169) tab.

Google Cloud Storage

- **JSON credentials:** The JSON credentials for the service account key used to access the Google Cloud Storage bucket.
For more information about JSON credentials, see the [Google Cloud online help](#).
- **Bucket name:** The name of the bucket in Google Cloud Storage where you want the data files to be written. This must be the same as the bucket you configured for your Databricks cluster.
- **Target directory:** Where to create the data files in the specified bucket.

Microsoft Azure Data Lake Storage (ADLS) Gen2

- **Storage account:** The name of your storage account.



To connect to an Azure resource on Government Cloud or China Cloud, you need to specify the full resource name of the storage account. For example, assuming the storage account is "myaccount", then the resource name for China Cloud would be `myaccount.dfs.core.chinacloudapi.cn`

In addition, you also need to specify the login URL using the `adlsLoginUrl` internal parameter. For China Cloud, this would be `https://login.chinacloudapi.cn`

For information on setting internal parameters, see *Setting advanced connection properties* (page 694)

- **Azure Active Directory Tenant ID:** The Azure Active Directory tenant ID.
- **Application Registration Client ID:** The application registration client ID.
- **Application Registration Secret:** The Application registration secret.
- **Container:** The container in which your folders and files and folders reside.
- **Target directory:** Specify where to create the data files on ADLS.



- The **Target folder** name can only contain ASCII characters.
- Connecting to a proxy server with a username and password is not supported with Azure Data Lake Storage (ADLS) Gen2 storage.

- **Databricks storage access method:** Choose which method your Databricks cluster uses to access the Amazon S3 storage: **Access Directly** (the default) or **Access through DBFS Mount**. AWS S3 storage can be accessed by mounting buckets using DBFS or directly. Replicate needs to know which method Databricks uses to access the storage so that it can set the "location" property when it creates the tables in Databricks. The "location" property enables Databricks to access the storage data using its configured storage access method.
 - **Mount Path** - When **Access through DBFS Mount** is selected, you also need to specify the mount path.



The mount path cannot contain special characters or spaces.

For more information on configuring Databricks to access Azure storage, see <https://docs.databricks.com/data/data-sources/azure/azure-storage.html>

Microsoft Azure Blob Storage



This storage type is not available when Replicate is installed on Linux.

- **Storage account:** The name of an account with write permissions to the container.



To connect to an Azure resource on Government Cloud or China Cloud, you need to specify the full resource name of the storage account. For example, assuming the storage account is `MyBlobStorage`, then the resource name for China cloud would be `MyBlobStorage.dfs.core.chinacloudapi.cn`

For information on setting internal parameters, see *Setting advanced connection properties* (page 694)

- **Access key:** The account access key.
- **Container name:** The container name.
- **Target directory:** Specify where to create the data files on Blob storage.
- **Databricks storage access method:** Choose which method your Databricks cluster uses to access the Amazon S3 storage: **Access Directly** (the default) or **Access through DBFS Mount**. AWS S3 storage can be accessed by mounting buckets using DBFS or directly. Replicate needs to know which method Databricks uses to access the storage so that it can set the "location" property when it creates the tables in Databricks. The "location" property enables Databricks to access the storage data using its configured storage access method.
 - **Mount Path** - When **Access through DBFS Mount** is selected, you also need to specify the mount path.



The mount path cannot contain special characters or spaces.

For more information on configuring Databricks to access Azure storage, see <https://docs.databricks.com/data/data-sources/azure/azure-storage.html>



All tables being replicated in a single task must be configured to access the storage using the **Databricks storage access method** defined in the endpoint settings (see earlier). The same is true for Replicate Control Tables, which are common to all tasks. Any tables that are configured to use a different storage access method will not be replicated or, in the case of Control Tables, will not be written to the target database. To prevent this from happening, you must perform the procedures below if you need to do any of the following:

- Change the Databricks storage access method during a task and retain the existing tables
- Define a new task with a Databricks target whose Databricks storage access method differs from existing tasks (with a Databricks target)

To change the Databricks storage access methods during a task:

1. Stop the task.
2. Change the **Databricks storage access method**.
3. For all tables (including Control Tables), execute the ALTER TABLE statement:
`ALTER TABLE table_identifier [partition_spec] SET LOCATION 'new_location'`
For details, see <https://docs.databricks.com/spark/latest/spark-sql/language-manual/sql-ref-syntax-ddl-alter-table.html#set-location>.
4. Start the task.

To create a new task with a Databricks endpoint whose "Databricks storage access method" differs from existing tasks:

- When you define the task, specify a dedicated Control Table schema in the task settings' Control tables (page 1169) tab.

Setting advanced connection properties

The table below describes the settings in the **Advanced** tab.

File format

If your cluster type is **SQL Warehouse**, then select either **Parquet** (the default) or **CSV**.

If your cluster type is **All-purpose**, then select either **Text** (the default), **Parquet** or **Sequence**.



When both **Use Unity Catalog** and **All-purpose clusters** are selected, the **Sequence** storage format will not be available. Choose **Text** or **Parquet** instead.

See information on setting the cluster type, see *Setting general connection properties* (page 688).

File attributes

- **Maximum file size:** Specify the maximum file size of each target file. When the data reaches the maximum size, the file will be closed and written to the specified target folder.

- **Compress files using:** Select the compression method to use in the selected storage.

Change processing

Apply/Store changes when:

- **File size reaches:** Specify the minimum size of the data required to create a file in idle state.
- **Elapsed time reaches:** Specify the maximum time to wait before applying the changes in idle state.



*To facilitate rapid delivery of DDL messages, files are uploaded immediately, regardless of the specified **File size reaches** or **Elapsed time reaches** values.*

Proxy Server



The options displayed depend on the selected storage type.

For information on the available storage types, see [Setting general connection properties \(page 688\)](#)

- **Use proxy server** - Select this option to connect via a proxy server, and then provide the proxy connection settings:
- Choose **Connect via proxy server to** options according to which resources you want to access via a proxy server:
 - **Database (via ODBC)**
 - **Cloud storage**



*When the **Cloud storage** option is selected and the storage type is either **Google Cloud Storage** or **Microsoft Azure Data Lake Storage (ADLS) Gen2**, proxy authentication is not supported.*

- **Azure Active Directory**



*Only available when **Microsoft Azure Data Lake Storage (ADLS) Gen2** is the selected storage type. When **Azure Active Directory** is selected, proxy authentication is not supported.*

- **Proxy connection:** Choose one of the following:
 - **Use endpoint proxy setting:** Choose this option to use the proxy settings defined for this endpoint.
 - **Use default proxy settings:** Choose this option to use the default proxy settings defined for all endpoints. The default settings are defined in the Replicate server settings' [Default Proxy Server](#) tab.
- **Host name:** The host name of the proxy server.
- **Port:** The port via which to access the proxy server.

- **User name:** The user name for the proxy server.
- **Password:** The password for the proxy server.
- **Scheme:** Select which protocol to use to access the proxy server (HTTP or HTTPS). In order to use HTTPS, you must first install the CA certificate that signed the proxy's certificate on the Replicate Server machine. Do this as follows:
 - **On Windows:** Add the CA certificate to the Trusted Root Certification Authorities store of Local Computer
 - **On Linux:** Add the CA certificate to `/etc/pki/tls/certs/ca-bundle.crt`



*The HTTPS scheme is only supported when both the **Amazon S3** storage type and the **Cloud storage** proxy option (only) are selected.*

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.11 Using a file as a target

This section describes how to set up and use delimited text files as a target in a replication task. You can use the File target endpoint to export database tables to files, which can then be used as a source in a Replicate task with a File source endpoint.

In this section:

- *General overview (page 196)*
- *File target overview (page 698)*
- *Limitations and considerations (page 699)*
- *Change data partitioning (page 700)*
- *File target data types (page 700)*
- *Setting general properties (page 701)*
- *Setting advanced connection properties (page 704)*
- *Generating reference files (page 708)*

File target overview

The File target endpoint generates the data files either in delimited text file format (e.g. CSV) or in JSON format (according to the format selected in the endpoint settings).

Delimited text files are used to store data in tabular format. Examples of delimited text file formats include the CSV (Comma Separated Values) and TSV (Tab Separated Values) formats. Some organizations may implement procedures that export data from a database to a delimited text file while others may simply prefer this format as a convenient way of storing tabular data.

In a delimited text file, each record in the table occupies a separate row. Delimiters are used to mark the beginning of a new row or the beginning of a new column. Virtually any character can be used as a delimiter, although a newline (\n) is often used to separate rows, and commas are commonly used to separate columns.

In JSON files, each record is represented by a single line.

So, for example, the following table:

JSON data format example

book_id	title	price	is_hardcover
123	Angels	6.99	false
456	The Fallen	6.49	true
789	Rise Up	7.23	true

Will be represented as:

```
{ "book_id": 123, "title": "Angels", "price": 6.99, "is_hardcover": false }
{ "book_id": 456, "title": "Fallen", "price": 6.49, "is_hardcover": true }
{ "book_id": 789, "title": "Rise Up", "price": 7.23, "is_hardcover": true }
```

When using a File as a target in a Replicate task, both the Full Load and the CDC data are written to CSV or JSON files (depending on the endpoint settings). While the explanations in this topic relate to CSV files, the same is true for JSON files.

Full Load files are named using incremental counters e.g. LOAD00001.csv, LOAD 00002.csv, etc. whereas Apply Changes files are named using timestamps e.g. 20141029-1134010000.csv.



When [Parallel Load](#) is used, the naming convention for Full Load files is slightly different:

`LOAD_$(SegmentID)_$(IncreasingCounter)`

Example:

```
LOAD_1_00000001 | LOAD_1_00000002 | LOAD_1_00000003 | LOAD_2_00000001
| LOAD_2_00000002
```



- The Apply Changes CSV files appear with a **.tmp** extension while they are in idle state. For more information on idle state, see [Change Processing](#).
- When the [Create metadata files in the target folder](#) option is enabled, a corresponding metadata file is created using the same naming format, but with a **.dfm** extension.

For each source table, a folder is created under the specified target folder. All files - i.e. Full Load, Apply Changes, and Metadata (if enabled) - are written to the relevant folder, according to the settings defined in the File target's **General tab**.

After a task completes, you can define another task with a File source endpoint that uses the generated CSV files.

DDL handling

When a DDL change is captured, Replicate will close the data file and also create a DFM file if the **Create metadata files in the target folder** option is enabled. When the next batch of changes arrives, Replicate will create a new data file containing the changes. Note that the DFM file created for the new data file will match the new table structure.

Limitations and considerations

The following limitations apply to the File target endpoint:

- Only the following DDLs are supported: Truncate table, Drop table, Create table, Add Column, Rename Column, Drop Column, and Convert Data Type.
- Full LOB Mode is not supported
- UPDATE and DELETE statements are not supported in Apply Changes replication mode
- Batch optimized apply mode is not supported
- Target lookup is not supported
- The <target folder> parameter cannot include special characters
- When the **When source table is altered** Apply Changes setting is set to [Ignore ALTER](#), data files will be updated, but metadata files and metadata headers (if defined) will not.
- Source schema and table names cannot contain periods. If they do, you should use a [global transformation](#) to remove/replace the periods; otherwise, data loss will occur.
- Database names, schema names, or table names containing slash (/) or backslash (\) characters are not supported.
- The following Control Tables are not supported as they require UPDATE/DELETE operations, which are not supported:
 - **Replication Status** (requires UPDATE).
Name on target: `attrep_status`
 - **Suspended Tables** (requires DELETE).
Name on target: `attrep_suspended_tables`
For more information on Control Tables, see *Control tables (page 1169)*.

Change data partitioning

When replicating to a File target, for each of the source tables, a directory is created under the specified target directory. When *Change Data Partitioning* (page 1177) is enabled, an additional sub-directory is created under the corresponding table directory. The data and metadata (when the metadata option is enabled) files are located in the partition subdirectory, as in the following example:

```

{Target Directory}
  {Table_1}
    {Partition_1}
      Data files
      DFM files
    {Partition_2}
      Data files
      DFM files
    {Partition_3}
      Data files
      DFM files
  {Table_2}
    {Partition_1}
      Data files
      DFM files
    {Partition_2}
      Data files
      DFM files
    {Partition_3}
      Data files
      DFM files
  
```

Information about the partitions is written to the **attrep_cdc_partitions** Control Table. For information about this table, see *Change data partitions* (page 1301).

File target data types

The following table shows the default mapping from Qlik Replicate data types to File target data types. Note that the data type mapping is only relevant if the [Create metadata files in the target folder](#) option is enabled.

For information on source data type mappings, see the section for the source endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).

Data types

Qlik Replicate Data Types	File Target Data Types
DATE	DATE
TIME	TIME
DATETIME	DATETIME
BYTES	BYTES (length)

Qlik Replicate Data Types	File Target Data Types
BLOB	BLOB
REAL4	REAL4 (7)
REAL8	REAL8 (14)
INT1	INT1 (3)
INT2	INT2 (5)
INT4	INT4 (10)
INT8	INT8 (19)
UINT1	UINT1 (3)
UINT2	UINT2 (5)
UINT4	UINT4 (10)
UINT8	UINT8 (20)
NUMERIC	NUMERIC (p,s)
STRING	STRING (Length)
WSTRING	STRING (Length)
CLOB	CLOB
NCLOB	NCLOB
BOOLEAN	BOOLEAN (1)

Setting general properties

This section describes how to configure general properties. For an explanation of how to configure advanced properties, see **Setting Advanced Properties** below.

To configure general properties for the File target endpoint:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.
2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the endpoint being used.
3. In the **Description** field, type a description that helps to identify the File endpoint. This is optional.
4. Select **Target** as the endpoint **role**.
5. Select **File** as the endpoint **Type**.
6. In the **Target folder** field, specify the full path of the folder to which you the target files to be written.
7. Configure the remaining settings in the **General** tab as described below.

File Format

Format

You can choose to create the target files in **CSV** or **JSON** format.

In a JSON file, each record is represented by a single line, as in the following example:

```
{ "book_id": 123, "title": "Alice in Wonderland", "price": 6.99, "is_hardcover": false }  
{ "book_id": 456, "title": "Winnie the Pooh", "price": 6.49, "is_hardcover": true }  
{ "book_id": 789, "title": "The Cat in the Hat", "price": 7.23, "is_hardcover": true }
```



*If you choose **JSON** format, the following fields will be hidden as they are only relevant to CSV format: **Field delimiter**, **Record delimiter**, **Null value**, **Quote character**, **Escape character**, **Code page**, and **Add metadata header**.*

About delimiters

Delimiters can be standard characters or a hexadecimal (hex) value. Note that the "0x" prefix must be used to denote a hexadecimal delimiter (e.g. 0x01 = SOH). In the **Field delimiter**, **Record delimiter** and **Null value** fields, the delimiter can consist of concatenated hex values (e.g. 0x0102 = SOHSTX), whereas in the **Quote character** and **Escape character** fields, it can only be a single hex value.



The hexadecimal number 0x00 is not supported (i.e. only 0x01-0xFF are supported).

Field delimiter

The delimiter that will be used to separate fields (columns) in the target files. The default is a comma.

Example using a comma as a delimiter:

```
"mike","male"
```

Record delimiter

The delimiter that will be used to separate records (rows) in the target files. The default is a newline (\n).

Example:

```
"mike","male"\n
```

```
"sara","female"\n
```

Null value

The string that will be used to indicate a null value in the target files.

Example (where \n is the record delimiter and @ is the null value):

```
"mike","male",295678\n
```

```
"sara","female",@\n
```

Quote Character

The character that will be used at the beginning and end of a text column. The default is the double-quote character ("). When a column that contains column delimiters is enclosed in double-quotes, the column delimiter characters are interpreted as actual data, and not as column delimiters.

Example (where a @ is the quote character):

```
@mike@,@male@
```

Escape character

The character used to escape a quote character in the actual data. The default is the double-quote character (").

Example (where " is the quote character and \ is the escape character):

```
1955,"old,\"rare\", Chevrolet", "$1000"
```

Add metadata header

When the target storage format is set to **Text**, you can optionally add a header row to the data files. The header row can contain the source column names and/or the intermediate (i.e. Replicate) data types.

Example of a target file with a header row when both **With column names** and **With data types** are selected:

```
Position:DECIMAL(38,0),Color:VARCHAR(10)
```

```
1,"BLUE"
```

```
2,"BROWN"
```

```
3,"RED"
```

```
...
```

Code page

Specify the code page of your target files if it is different from the default (65001).



Windows and Linux systems use different code page conventions. The specified code page must comply with the code page convention of the source file system.

File Attributes

- **Maximum file size** - The maximum size a file can reach before it is closed (and optionally compressed). This value applies both to data files and to Reference Files. For information on generating reference files, see *Generating reference files (page 708)*.
- **Compress files using** - Choose **GZIP** to compress the target files or **NONE** (the default) to leave them uncompressed.

Change Processing

- **Consider state idle when no changes have been processed for** - Specify how long to wait before considering the state to be idle. In idle state, you can apply changes to files using data that has already been processed if the specified size and time conditions are met (see below).
- **Apply/store changes when:**
 - **File size reaches** - Specify the maximum size of the data required in order to apply changes to the target file in idle state.
 - **Elapsed time reaches** - Specify the maximum time to wait before applying the changes in idle state.
- **Allow a single transaction to be split into multiple files** - By default, a single transaction will **not** be split across multiple files, regardless of the values specified in the **File size reaches** and **Elapsed time reaches** fields. This is important for organizations who require files to contain transactions in their entirety. However, this may also result in very large file sizes. For example, if the **File size reaches** value is **32 MB** and Replicate starts to apply changes for a new **2 GB** transaction at **31 MB**, the target file will only be closed at **2.031 GB**.
You should therefore select this option if it is critical that the values in the **File size reaches** and **Elapsed time reaches** fields are adhered to (even if it means splitting a transaction across multiple files).

Metadata Files

Create metadata files in the target folder - When this option is selected, for each data file, a matching metadata file with a **.dfm** extension will be created under the specified target folder. The metadata files (which are in standard JSON format) provide additional information about the task/data such as the source endpoint type, the source table name, the number of records in the data file, and so on.

For a full description of the metadata file as well as possible uses, see *Metadata file description (page 1310)*.



To verify that the connection information you entered is correct, click **Test Connection**.



If the connection is successful a message in green is displayed. If the Replicate is unable to connect to the endpoint, an error message will be displayed at the top of the dialog box.



Setting advanced connection properties

In the **Advanced** tab, you can enable the creation of a reference file and set post-processing actions. These options are described in detail below.

10 Adding and managing target endpoints

Advanced tab settings

Option	Description
Generate reference files	<p>Select this option to generate a Reference File containing the full path to the Apply Changes data files.</p> <div data-bbox="408 427 1390 562" style="border: 1px solid #ccc; padding: 5px;"><p> <i>The reference file only points to the location of the Apply Changes files, and not the Full Load files.</i></p></div> <p>For more information on this feature, see <i>Generating reference files (page 708)</i>.</p> <p>For information on using reference files with the File source endpoint, see <i>Reference files (page 197)</i>.</p>
Reference file folder	<p>The folder on the Replicate machine in which the Reference File will be created.</p> <div data-bbox="408 801 1390 981" style="border: 1px solid #ccc; padding: 5px;"><p> <i>Multiple tasks with file target endpoints that have the same target directory are not supported (as each task will attempt to write to the same reference file).</i></p></div> <p>Example:</p> <p>c:\temp\</p>

Option	Description
Post-process files	<p>You can process the final target files using a custom command. The command will be run whenever a data file is created.</p> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p> If the Generate a reference file option is selected, a row (specifying the file's location) will be added to the Reference File only after the command completes successfully.</p> </div> <ul style="list-style-type: none"> • Command name - The location of the command e.g. <code>C:\utils\move.exe</code>. • Working directory - The directory where you want the command to run. • Parameters - Specify any parameters that need to be passed to the command during runtime. You can use the following built-in parameters: <ul style="list-style-type: none"> <code>\${FILENAME}</code> - The full path to the CSV file containing the full load or CDC data. <code>\${METADATA_FILENAME}</code> - The full path to the DFM file containing the metadata. <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <ul style="list-style-type: none"> •  If the CSV/DFM file paths contain spaces, you must enclose these parameters with quotation marks (e.g. <code>"\${FILENAME}"</code>). • The post-processing command must return a proper exit code. For information on the available exit code options, see <i>Exit codes for the Post-Processing command (page 707)</i>. </div> <p>For information on creating metadata files, see <i>Setting general properties (page 701)</i>.</p>
After post processing completes	<p>You can decide what to do with the original target files after post-processing completes:</p> <ul style="list-style-type: none"> • Do nothing - Leaves the files in their original location • Delete files - Deletes the files from the disk • Archive files to folder - Archives the files to the specified folder.

Exit codes for the Post-Processing command

The post-processing command must return a proper exit code. You can either use the standard exit code values described below or set a custom exit code value as described in **Setting Post Command exit codes with an internal parameter** below.

Standard Post Command exit codes

- **0** - Success
- **1** - Recoverable error. The task will recover from the point of failure according to the settings in the *Environmental errors (page 1187)* tab.
- **2** - Table error. If a table error occurs, Replicate will handle the error according to the settings in the *Table Errors (page 1191)* tab.
- **3** (or any other value e.g. -100) - Fatal error. The task will fail and not attempt recovery.

Setting Post Command Exit Codes with an Internal Parameter

You can use internal parameters to set exit codes with custom values. This is especially useful if your application already uses the standard exit code values.

See **Standard Post Command exit codes** above for a description of the exit codes.

- `successExitCode`
- `recoverableErrorExitCode`
- `tableErrorExitCode`
- `fatalErrorExitCode`

Internal parameters

Internal parameters are parameters that are not exposed in the UI and should only be used if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your setting by clicking the View Setting Summary link. This is useful if you need to send a summary of your setting to Qlik Support.

Generating reference files

In the **Advanced** tab of the File Target endpoint, you can enable the **Generate a reference file** option. The Reference File contains a list of the [Change File](#) locations and is therefore only relevant if the task's Apply Changes or Store Changes options are enabled.

The reference file name format is as follows:

```
<file_target_endpoint_name><counter>.csv|json
```

Example:

```
FileTarget00000001.csv
```



The counter suffix increases incrementally each time a new Reference File is generated (i.e. when the file reaches the maximum size defined [in the General tab](#)). Once a new Reference File has been generated, you can delete the old reference file(s) if required.

Whenever an Apply Changes data file is created, a new row is added to the Reference File in the following format:

```
<Source_Table_Name>,<full_path_to_data_file>
```

Example:

```
MyTable,c:\temp\filetarget\dbo.MyTable\20170102-091759447.csv
```

Note that if the **Post-process files** option in the **Advanced** tab is also enabled, the Reference File will be generated *after* the post-processing completes.



- When both the **Post-process files** and the **Delete files** (after post-processing completes) options are enabled, the reference file will not be generated.
- If the **Archive files to folder** (after post-processing completes) option is selected, the reference file will be updated to reflect the archive location of the data files.

10.12 Using Google Cloud AlloyDB for PostgreSQL as a target

This section describes how to set up a Google Cloud AlloyDB for PostgreSQL target endpoint in a replication task.


In this section:

- *Prerequisites (page 710)*
- *Limitations (page 711)*
- *Security requirements (page 711)*
- *Supported PostgreSQL data types (page 711)*
- *Setting general connection properties (page 712)*
- *Setting advanced connection properties (page 714)*

Prerequisites

This topic describes the client prerequisites when replicating to a Google Cloud AlloyDB for PostgreSQL target.

Qlik Replicate Server for Windows

- **The PostgreSQL ODBC Driver:** PostgreSQL ODBC Driver [psqlodbc_13_01_0000-x64](#) or above must be installed on the Qlik Replicate machine.
- **PgAdmin:**
 1. Download and install  [PgAdmin 4 \(Windows\)](#) on the Qlik Replicate machine.
 2. Add the "psql.exe" path to the system PATH.
- Microsoft Visual Studio 2013 Redistributable (x86) - *not* (x64) - must be installed on the Qlik Replicate machine.

Qlik Replicate Server for Linux

Download PostgreSQL 13 ODBC driver, and client programs and libraries.

- **Example PostgreSQL ODBC file name:** postgresql13-odbc-13.02.0000-1PGDG.rhel8.x86_64.rpm
- **PostgreSQL client programs and libraries file name:** postgresql13-13.8-1PGDG.rhel<version>.x86_64.rpm

Then, open a shell prompt and do the following:

1. Stop the Replicate service:
`./areplicate stop`
2. Optionally, confirm that the service has stopped:
`./areplicate status`
3. Install PostgreSQL 13 client programs and libraries.
4. Install the ODBC driver for PostgreSQL 13.
5. Change the working directory to `<Replicate-Install-Dir>/bin`.
6. Copy the driver location to the `site_arep_login.sh` file:
`echo "export LD_LIBRARY_PATH=\$LD_LIBRARY_PATH:/usr/lib:/usr/lib64:/usr/pgsql-13/lib" >> site_arep_login.sh`
This will add the driver to "LD_LIBRARY_PATH" and update the driver location in the **site_arep_login.sh** file.
7. Optionally, confirm that the driver location was copied:
`cat site_arep_login.sh`
8. Makes sure that the `/etc/odbcinst.ini` file contains an entry for PostgreSQL, as in the following example:

```
[PostgreSQL Unicode(x64)]
Description = PostgreSQL ODBC driver
Driver = /usr/pgsql-13/lib/psqlodbc.so
Setup = /usr/pgsql-13/lib/psqlodbcw.so
UsageCount = 1
```
9. Start the Replicate service:
`./areplicate start`
10. Optionally, confirm that the service has started:
`./areplicate status`

Limitations

UPDATE and DELETE operations on tables that contain duplicate records, but do not have a Primary Key/Unique Index are not supported. Any changes to these tables will affect all the records in the target.

Security requirements

The user specified in the **General** tab when *Setting general connection properties (page 712)* must be a registered user in the Google Cloud AlloyDB for PostgreSQL database.

Supported PostgreSQL data types

The Google Cloud AlloyDB for PostgreSQL endpoint for Qlik Replicate supports most PostgreSQL database data types. The following table shows the PostgreSQL database target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source database you are using. For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

Data types

Qlik Replicate data types	PostgreSQL data types
BOOL	BOOL
BYTES	BYTEA
DATE	DATE
TIME	TIME
DATETIME	If scale is => 0 and =< 6, then: TIMESTAMP If scale is => 7 and =< 12, then: VARCHAR (37)
INT1	SMALLINT
INT2	SMALLINT
INT4	INTEGER
INT8	BIGINT
NUMERIC	DECIMAL (P, S)
REAL4	FLOAT4
REAL8	FLOAT8

Qlik Replicate data types	PostgreSQL data types
STRING	If length is 1 - 21845, then: VARCHAR (Length in Bytes = The STRING value multiplied by three) If length is 21846 - 2147483647, then: VARCHAR (65535)
UINT1	SMALLINT
UINT2	INTEGER
UINT4	BIGINT
UINT8	BIGINT
WSTRING	If length is 1 - 21845, then: VARCHAR (Length in Bytes = The WSTRING value multiplied by three) If length is 21846 - 2147483647, then: VARCHAR (65535)
BLOB	BYTEA
NCLOB	TEXT
CLOB	TEXT

Data types when replicating from a PostgreSQL source

When replicating from a PostgreSQL source, the target table will be created with the same data types for all columns, apart from columns with user-defined or PostGIS data types. In such cases, the data type will be created as "character varying" in the target.

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a Google Cloud AlloyDB for PostgreSQL target endpoint to Qlik Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the PostgreSQL database. This is optional.
4. Select **Target** as the database **role**.
5. From the **Type** drop-down list, select **Google Cloud AlloyDB for PostgreSQL**.

10 Adding and managing target endpoints

6. Type the **Host** name. This is the name or IP address or host name of the computer with the PostgreSQL database that you want to access.
7. Optionally, change the default port (5432).
8. Enter the PostgreSQL database authentication information (**Username, Password**) of an authorized PostgreSQL user. If you do not know this information, see your PostgreSQL database system manager.



Consider the following:

- *This information is case sensitive.*
- *To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**. If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box. To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.*



Make sure that the specified PostgreSQL database user has the correct access privileges.

9. Type the **Database name**.

SSL authentication options

In the **Security** section, you can configure Replicate to connect to the PostgreSQL database using SSL.



The specified certificates are relevant only for bulk insert operations.

- **SSL Mode:** Select one of the following:
 - **Disable** - Connect with a surname and password only.
 - **Allow** - Establish an encrypted connection if requested by the server.
 - **Prefer** - Establishes an encrypted connection if the server supports encrypted connections, falling back to an unencrypted connection if an encrypted connection cannot be established.
 - **Require** - Establishes an encrypted connection if the server supports encrypted connections. The connection attempt fails if an encrypted connection cannot be established.
 - **Verify CA** - Similar to **Required**, but also verifies the server Certificate Authority (CA) certificate against the configured CA certificates. The connection attempt fails if no valid matching CA certificates are found.
 - **Verify Full** - Similar to **Verify CA**, but also performs host name identity verification by checking the host name the client (i.e. Replicate) uses for connecting to the server against the identity in the certificate that the server sends to the client. The client checks whether the host name that it uses for connecting matches the Common Name value in the server certificate. The connection fails if there is a mismatch.
- **Client certificate path** - The path to the client certificate requested by the server.

- **Secret key path** - The path to the client private key file in PEM format.
- **CA path** - The path to the Certificate Authority (CA) that issued the client certificate file in PEM format.
 - **CRL path** - The path to the CRL certificate. This file contains certificates revoked by certificate authorities. If the server certificate appears in this list, the connection will fail.
 - **SSL compression** - Select this option to compress the data before it is encrypted.

Setting advanced connection properties

In the **Advanced** tab, you can configure additional options and see a summary of your settings.

- **Max file size (KB)**: Select or type the maximum size (in KB) the CSV file can reach before it is loaded into the Google Cloud AlloyDB for PostgreSQL target database. The default value is 32,000 KB.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.13 Using Google Cloud BigQuery as a target

This section describes how to set up and use a Google Cloud BigQuery target endpoint in a replication task.

In this section:

- *Prerequisites (page 716)*
- *Limitations and considerations (page 717)*
- *Supported data types (page 718)*
- *Setting general connection properties (page 719)*
- *Setting advanced connection properties (page 720)*

Prerequisites

The following section describes the client prerequisites when replicating to a Google Cloud BigQuery target.

Permissions

This section describes the Google Cloud BigQuery Service Account permissions required by Replicate.

When you create a service account key, a JSON file containing the connection information will be downloaded to your computer. You will need to copy the contents of this file to the **Service account key** field in the Google Cloud BigQuery endpoint.

For additional information about dataset creation, see *Setting advanced connection properties (page 720)*.

Permissions required to create the target dataset automatically

If you want the dataset to be created automatically by Replicate, the following permissions are required:

BigQuery > BigQuery Job User

BigQuery > BigQuery Data Editor


Permissions required when target dataset already exist

If the dataset already exists, you need to perform the following steps:

1. Create a service account with the following permission:
BigQuery > BigQuery Job User
2. Navigate to the dataset that you want Replicate to use, and then:
 - a. Add the service account you just created as a principal.
 - b. Assign the role: **BigQuery Data Editor**.


Drivers

Qlik Replicate on Windows

1. Download  [Simba ODBC driver version 3.0.0.1001](#) and install it on the Replicate Server machine:
2. Install Google Cloud SDK 402.0.0 on the Replicate Server machine as follows:
 - a. Download and install the latest Google Cloud SDK version from Google.
 - b. Downgrade to version 402.0.0 by issuing the following command:
`gcloud components update --version 402.0.0`

Qlik Replicate on Linux

To be able to use Google Cloud BigQuery as a target endpoint in a Replicate task when Replicate is running on Linux, you need to perform the following procedure:

1. Download  [Simba ODBC driver version 3.0.0.1001](#) to the Replicate machine.
2. Extract:
SimbaODBCDriverforGoogleBigQuery64_3.0.0.1001.tar.gz
to:
`/opt/SimbaODBCDriverforGoogleBigQuery_3.0.0.1001-Linux`

3. Move the files **GoogleBigQueryODBC.did** and **simba.googlebigqueryodbc.ini** to the **lib** directory under the Simba ODBC driver directory.
Example:
Move the files from:
`/opt/SimbaODBCDriverforGoogleBigQuery_3.0.0.1001-Linux/setup`
to:
`/opt/SimbaODBCDriverforGoogleBigQuery_3.0.0.1001-Linux/SimbaODBCDriverforGoogleBigQuery64_3.0.0.1001/lib`
4. Edit the **simba.googlebigqueryodbc.ini** file as follows:
 - a. Change the `ErrorMessagesPath` to the path of the XML file containing the ODBC messages. The default location is:
`ErrorMessagesPath=/opt/SimbaODBCDriverforGoogleBigQuery_3.0.0.1001-Linux/SimbaODBCDriverforGoogleBigQuery64_3.0.0.1001/ErrorMessages`
 - b. Change the `DriverManagerEncoding` to UTF-16.
5. Add the following path to the **site_arep_login.sh** file located in the Replicate **bin** directory:
`export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/SimbaODBCDriverforGoogleBigQuery_3.0.0.1001-Linux/SimbaODBCDriverforGoogleBigQuery64_3.0.0.1001/lib`
6. Edit the **/etc/odbcinst.ini** file and add the driver path (i.e. the path where the driver is installed):

```
[ODBC Drivers]
Simba= Installed
Simba ODBC Driver for Google BigQuery = Installed
[Simba ODBC Driver for Google BigQuery]
Description=Simba ODBC Driver for Google BigQuery(64-bit)
Driver=/opt/SimbaODBCDriverforGoogleBigQuery_3.0.0.1001-Linux/SimbaODBCDriverforGoogleBigQuery64_3.0.0.1001/lib/libgooglebigqueryodbc_sb64.so
```
7. Change to the Replicate **bin** directory (`<install_dir>/bin`). Then stop and start the Replicate service:
`./areplicate stop`
`./areplicate start`
8. Download and install [Google Cloud SDK 402.0.0 for Linux](#) on the Replicate machine.

Limitations and considerations

The following limitations apply when using Google Cloud BigQuery as a target in a Replicate task:

- The following DDLs are not supported:
 - Drop column
 - Rename column
 - Change column data type
 - Rename table
- Files will be loaded to BigQuery uncompressed and unencrypted.
- The [Transactional apply Change Processing Mode](#) is not supported.
- The [Allow unlimited LOB size](#) LOB handling option is not supported.

- Capturing changes from source tables without a Primary Key or Unique Index is not supported. If you need to capture changes from such tables, you can add a Primary Key [using a transformation](#). Additionally, Primary Key or Unique Index columns cannot contain NULLs. If you are aware that such columns will be populated with NULL values, [define a transformation](#) to change the NULLs to non-nullable values.
- The **Log record to the exceptions table** error handling option for Apply Conflicts and Data Errors is not supported.
- With the following task settings configuration, Replicate will perform a DELETE operation followed by an INSERT operation when an UPDATE is encountered:
 - [Change Processing Mode](#) is set to **Batch optimized apply**
 - The **Apply changes using SQL MERGE** option is *not* selected
 - The [Apply Conflicts error handling](#) for UPDATES is set to: **No record found for applying an UPDATE: INSERT the missing target record**

As Google Cloud BigQuery does not support rollback, in the event that Replicate is unable to insert the updated row, the data will be deleted from the target.

Supported data types

The following table shows the Google Cloud BigQuery database target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source database you are using. For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

Qlik Replicate and Google Cloud BigQuery database data types

Qlik Replicate Data Types	Google Cloud BigQuery database data types
BYTES	BYTES
DATE	DATE
TIME	TIME
DATETIME	If fraction <=6: TIMESTAMP If fraction >6: STRING
INT1	INTEGER
INT2	INTEGER
INT4	INTEGER
INT8	INTEGER

Qlik ReplicateData Types	Google Cloud BigQuery database data types
NUMERIC	If the data can be stored in 38,9, then: NUMERIC If not, then: STRING with length of original precision +2.
REAL4	FLOAT
REAL8	FLOAT
STRING	STRING
UINT1	INTEGER
UINT2	INTEGER
UINT4	INTEGER
UINT8	INTEGER
WSTRING	STRING
BLOB	BYTES
NCLOB	STRING
CLOB	STRING
BOOLEAN	BOOLEAN

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a Google Cloud BigQuery target endpoint to Qlik Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Google Cloud BigQuery database. This is optional.
4. Select **Target** as the database **role**.
5. From the **Type** drop-down list, select **Google Cloud BigQuery**.
6. In the **Service account key** field, paste the contents of the JSON file (including curly brackets) that was downloaded when you created your BigQuery service account key.
7. To verify that the connection information you entered is correct, click **Test Connection**.

If the connection is successful a confirmation will be displayed at the top of the dialog box. If Replicate is unable to connect to the endpoint, an error message will be displayed instead.

Setting advanced connection properties

In the **Advanced** tab, you can set the following properties:

You can set the following properties:

- **Max file size (KB):** Select or type the maximum size (in KB) of a CSV file before the file is loaded into the Google Cloud BigQuery target database.
- **Use compression:** When selected, the endpoint will compress (using gzip) the CSV files before uploading them to Google Cloud BigQuery.
- **Dataset creation**
 - **Create datasets automatically** - Replicate will create the datasets automatically.



Create datasets automatically requires a higher permission level than **Do not create datasets**. For more information, see *Prerequisites* (page 716).

- **Do not create datasets** - Assumes the required datasets already exist on the target. When using this option, you must do one of the following:
 - Enter the name of the existing dataset in the task settings' [Target table schema](#) field.
 - OR-
 - Enter the name of the existing dataset in the following two places:
 - The table settings' [Table Schema](#) field.
 - The **Create target control tables in schema** field in the task settings' [Control Tables](#) tab.
- **Location:** Where to upload the dataset created by Replicate. If you choose **Other**, set the location code in the **Region code** field. Region codes are listed as "Region keys" in <https://cloud.google.com/bigquery/docs/locations>.
- **Default dataset prefix:** Specify a prefix for the BigQuery dataset where you want Replicate to create and update the target tables.



This is required when the Apply Changes replication option is selected.

- **Create tables as clustered by primary key:** When this option is selected, the target tables will be created as clustered (according to the first four Primary Key columns that support clustering). In general, clustered tables usually provide much better query performance as well as reducing billing costs.



- *Only tables with primary keys of the following data types can be created as clustered: DATE, BOOL, GEOGRAPHY, INT64, NUMERIC, STRING, and TIMESTAMP.*
- *If you select this option and then later decide that you do not want the tables to be created as clustered on the target, turn off this option and reload the task.*

- **Use proxy server** - Select this option to connect via a proxy server, and then provide the proxy connection settings:
- **Proxy connection:** Choose one of the following:
 - **Use endpoint proxy setting:** Choose this option to use the proxy settings defined for this endpoint.
 - **Use default proxy settings:** Choose this option to use the default proxy settings defined for all endpoints. The default settings are defined in the Replicate server settings' [Default Proxy Server](#) tab.
- **Host name:** The host name of the proxy server.
- **Port:** The port via which to access the proxy server.
- **User name:** The user name for the proxy server.
- **Password:** The password for the proxy server.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.14 Using Google Cloud Pub/Sub as a target

This section describes how to set up and use Google Cloud Pub/Sub as a target endpoint in a replication task.

In this section:

- *Overview (page 723)*
- *Prerequisites (page 724)*
- *Limitations (page 724)*
- *Supported data types (page 725)*
- *Setting general connection properties (page 726)*
- *Setting advanced connection properties (page 731)*
- *Data messages (page 732)*

Overview

The following topic provides an overview of the Google Cloud Pub/Sub endpoint.

Topic handling and message ordering

In a task configured with the Google Cloud Pub/Sub endpoint, Replicate acts as the Publisher. Messages can be published to separate topics for each table or to a single topic according to the endpoint settings. When messages are published, regardless of whether they are published to separate topics or to a single topic, Replicate creates a default subscription for each topic it creates if no subscription exists. The default subscription name is the topic name appended with the **-sub** suffix (for example **mytopic-sub**). Additionally, the ability to drop a topic (and its associated subscription) is only relevant when messages are published to separate topics and only if the following task settings are configured as follows:

- Full Load setting: If target table already exists: **DROP and CREATE table**
- Apply Changes setting: When source table is dropped: **DROP target table**

The endpoint provides a **Keep message changes in order** option which ensures that messages are published in the order they occurred. It also ensures that consumers receive the messages in the correct order by enabling the **Order messages with an ordering key** option when Replicate creates a default subscription. However, as mentioned above, Replicate only creates a default subscription if no subscription exists. Customers can of course create their own topics and subscriptions, in which case they will need to enable the **Order messages with an ordering key** option (and make sure the endpoint's **Keep message changes in order** option is enabled) in order to receive the messages in the correct order. For more information, see [Ordering messages](#). Note that Replicate turns on this option when it automatically creates the subscription (see above).

Transaction processing from a consumer perspective

When configuring the Google Cloud Pub/Sub endpoint, users can determine where messages are published.

During a task's CDC stage, committed changes that are detected by the Qlik Replicate source endpoint are grouped by transaction, sorted internally in chronological order, and then propagated to the relevant topics.

Each CDC message has both a Transaction ID as well as change sequence. As the change sequence is a monotonically growing number, sorting events by change sequence always achieves chronological order. Grouping the sorted events by Transaction ID then results in transactions containing chronologically sorted changes.

However, as Google Cloud Pub/Sub is a messaging infrastructure, applying changes is not feasible. The Google Cloud Pub/Sub endpoint, therefore, takes a different approach, which is to report all transactional events as messages.

If maintaining transaction consistency is important for the consumer implementation, it means that although the Transaction ID exists in all data messages, the challenge is to gather the messages in a way that facilitates identifying a whole transaction. An additional challenge is getting the transaction in the original order they were committed, which could be an even greater challenge if transactions are spread across multiple topics.

The simplest way of achieving the above goal is to direct Replicate to a specific topic (in the endpoint settings). This means that all data messages will end up in a single topic, thus guaranteeing ordered delivery both of transactions *and* of changes within a transaction. The consuming application could then consume messages - accumulating a transaction in some intermediate memory buffer - and when a new Transaction ID is detected, mark the previous transaction as completed.

Prerequisites

Make sure you fulfill the following prerequisites before using the Google Cloud Pub/Sub endpoint in a replication task.

Permissions

Before you can use Google Cloud Pub/Sub as a target endpoint in a Replicate task, you need to set the following permissions:

Topics

- `pubsub.topics.create` on the containing Cloud project (only required if you want Replicate to create topics when no topics exists)
- `pubsub.topics.delete` on the requested topic
- `pubsub.topics.get` on the requested topic
- `pubsub.topics.list` on the requested Cloud project
- `pubsub.topics.publish` on the requested topic

Subscriptions

- `pubsub.subscriptions.create` on the containing Cloud project (only required if you want Replicate to create a default subscription per topic when no subscription exists)
- `pubsub.subscriptions.delete` on the requested subscription
- `pubsub.subscriptions.get` on the requested subscription
- `pubsub.subscriptions.list` on the requested Cloud project

Authentication

You can authenticate using a service account key or using Application Default Credentials (ADC). You need to set up the desired authentication method before configuring the endpoint.

- **Service account key (JSON):** For information on creating a service account key, see [🔗 Create and delete service account keys](#)
- **Application Default Credentials (ADC):** For information on setting up Application Default Credentials, see [🔗 How Application Default Credentials works](#)

Limitations

When defining a task with Google Cloud Pub/Sub as the target endpoint, the following limitations apply:

- The Google Cloud Pub/Sub target endpoint does not support unlimited LOB size. Therefore, when replicating from source tables with LOB columns, *do not* select the **Allow unlimited LOB size** option. For more information on defining LOB settings, see *Target metadata (page 1166)*.

10 Adding and managing target endpoints

- **Batch optimized apply** mode is not supported. If this mode is set, the task will automatically switch to **Transactional apply** mode and a message will be written to the task's log file. For more information on these modes, see *Change Processing Tuning (page 1181)*.
- Store Changes mode is not supported.
- Google Cloud Pub/Sub topic names must adhere to the guidelines provided in [📄 Create and manage topics](#).
If the source table names exceed the maximum permitted length or contain unsupported characters, you need to do one of the following:
 - Modify the names before starting the task.
 - Define a table-level transformation if only a single table name needs to be changed.
 - Define a global transformation rule for multiple tables. For information on defining global transformations, see *Starting the Global Transformation Rules wizard (page 1107)*.
- The [Ignore ALTER](#) Apply Changes setting is not supported for changes to source data types and table renaming.
- Column names must adhere to the guidelines provided in [📄 Create and manage topics](#).
If a source column name does not adhere to this rule, then [a transformation should be used](#) to rename the column.
- The [Drop and Create table](#) **Target Table Preparation** option is only supported when the endpoint is configured to create a separate topic for each table. When the **Drop and Create table** option is set, the associated subscription will also be deleted.
- The [Truncate before loading](#) **Target Table Preparation** option is not supported.
- The Change Data Partitioning [Control table](#) is not supported.

Supported data types

The table below shows the default mappings from Qlik Replicate data types to JSON and Avro logical data types.

For information on source data type mappings, see the section for the source endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.



When using the JSON message format, binary values are represented as hexadecimal digits.

Data type mappings

Qlik Replicate Data Types	JSON	Avro Logical Data Types
DATE	STRING	DATE Annotates an Avro INT.
TIME	STRING	TIME-MILLIS Annotates an Avro INT.

Qlik Replicate Data Types	JSON	Avro Logical Data Types
TIMESTAMP	STRING	TIMESTAMP-MICROS Annotates an Avro LONG.
STRING	STRING	STRING
WSTRING	STRING	STRING
CLOB	STRING	STRING
NCLOB	STRING	STRING
NUMERIC	STRING	DECIMAL (p,s) Annotates an Avro BYTES.
BYTES	BYTES	BYTES
BLOB	BYTES	BYTES
REAL4	FLOAT	FLOAT
REAL8	DOUBLE	DOUBLE
INT1	INT	INT
INT2	INT	INT
INT4	INT	INT
UINT1	INT	INT
UINT2	INT	LONG
UINT4	LONG	LONG
INT8	LONG	LONG
UINT8	STRING	DECIMAL (20,0) Annotates an Avro BYTES.
BOOLEAN	BOOLEAN	BOOLEAN

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To define the general connection properties:

1. Click the **Manage Endpoint Connections** toolbar button.
The **Manage Endpoints Connections** dialog box opens.
2. Click the **New Endpoint Connection** toolbar button.
The **Name, Description, Type** and **Role** fields are displayed on the right.
3. In the **Name** field, specify a display name for the endpoint.

4. In the **Description** field, optionally type a description for the Google Cloud Pub/Sub endpoint.
5. Select **Target** as the endpoint **Role**.
6. Select **Google Cloud Pub/Sub** as the endpoint **Type**.
The dialog box is divided into **General** and **Advanced** tabs.
7. Continue as described below.

Access details

In the **Access Details** section, the following options can be set:

- **Project ID:** The project ID, which is available [from the Google Cloud console](#). For example, **mypubsub-227813**.
- **Authentication:** Select one of the following as appropriate:
 - **Service account key (JSON):** Paste the contents of the JSON file (including curly brackets) that was downloaded when you created your service account key.
For information on creating a service account key, see [Create and delete service account keys](#)
 - **Application Default Credentials (ADC):** Select this option to authenticate using preconfigured settings. For more information, see [How Application Default Credentials works](#)
- **Regional endpoint:** Select the regional endpoint corresponding to the subscribers' region in order to reduce egress (data retrieval) costs. If the endpoint does not appear in the list, select **Other** and then specify the region code in the **Other region (gRPC service)** field.
 - **Other region (gRPC service):** When the regional endpoint is **Other**, enter the region code. The region code should be formatted as shown in the **REST/HTTP endpoint** column in the [List of regional endpoints](#), but without the `https://`.
Example:
`us-newregion1-pubsub.googleapis.com`

Message properties

In the **Message Properties** section, set the following properties:

- **Format:** Choose **JSON** or **Avro**.
- **Compression:** Select **GZIP** (the default) or **None**.
- **Keep message changes in order:** When this option is selected, subscribers will get messages in the order they occurred. This is facilitated by a unique "Ordering Key" value being assigned to each message. As this might impact performance, subscribers can also determine the message order using the Transaction Last Event and Transaction Event Counter (which can be included in the message metadata).
- **On source update, include before-image:** When this option is selected, both pre-update data and post-update data will be included in the message.
- **Include metadata:** You can include the following metadata **In message only**, **In message attributes only**, or **In both message data and message attributes**.
 - **Schema Name** - The default is to include this metadata **In message attributes only**.
 - **Table Name** - The default is to include this metadata **In message attributes only**.

- **Operation** - The default is to include this metadata **In message attributes only**.
- **Change Sequence** - The default is to include this metadata **In message attributes only**.
- **Timestamp** - The default is to include this metadata **In message attributes only**.
- **Transaction ID** - The default is to include this metadata **In message attributes only**.
- **Change Mask** - The default is **Do not include**.
- **Transaction Event Counter** - The default is **Do not include**.
- **Transaction Last Event** - The default is **Do not include**.

For a description of the metadata, see *Data messages* (page 732).

Data message publishing

In the **Publish data to** field, choose one of the following:

- **Specific topic:** Select this option to publish the data to a single topic. Then, in the **Topic** field, either type a topic name or use the browse button to select a topic.
- **Separate topic for each table:** Select this option to publish the data to multiple topics corresponding to the source table names.
The target topic name consists of the source schema name and the source table name, separated by a period (e.g. "dbo.Employees").

Metadata message publishing

In the **Metadata Message Publishing** section, you can configure Schema Registry connection properties and Schema Registry subject properties.



This section is not relevant when the message format is JSON.

Schema Registry connection properties

- **Schema Registry servers:** Specify one or more Schema Registry servers using the following format (for high availability):
server1:port1[,server2[:port2]]

Example:

192.168.1.100:8081,192.168.1.101:8081
Replicate will connect to the first available host.

- **Use SSL:** Select this option to encrypt the data between the Replicate machine and the Schema Registry server(s). If the servers re configured to require SSL, then you *must* select this option.
 - **CA path:** Specify one of the following:
 - The full path (i.e. including the file name) to a specific CA certificate in PEM format
 - The directory containing certificate files with hash names

- **Authentication** - Select one of the following Schema Registry authentication options:
 - **None** - No authentication.
 - **Certificate** - Select to authenticate using a certificate.
If you select this option, you also need to provide the following information:
 - **Public key file** - The full path to the public key file on the Replicate Server machine.
 - **Private key file** - The full path to the private key file on the Replicate Server machine.
 - **Private key password** - The password for the private key file.
 - **Username and password** - Select to authenticate with a username and password.
- **Use proxy server** - Select to publish to the Schema Registry via a proxy server.
 - **Host name** - The host name of the proxy server.
 - **Port** - The port via which to access the proxy server.
 - **Scheme** - Select which protocol to use to access the server (**HTTP** or **HTTPS**). In order to use **HTTPS**, you must also specify the **SSL CA Path**.
 - **SSL CA Path**: The location of the CA file on the Replicate Server machine when **HTTPS** is the selected Scheme.

Schema Registry subject properties

From the **Subject name strategy** drop-down list, optionally change the default subject name strategy:

- Schema and Table Name (Default)
- Topic Name



- The default strategy (**Schema and Table Name**) is a proprietary Qlik strategy while **Topic Name** is a standard Confluence subject name strategies.
- If you select **Topic Name**, be aware of the following considerations:
 - When the "add \$topic column" method is used, the subject will be created only once (as the \$topic expression might create multiple subjects). See also steps 3 and 4 in *Overriding the default settings* (page 730).
 - The "Metadata only" [Advanced run](#) option is not supported. This is because Replicate depends on the arrival of the first record per table in order to create the subject.

For more information on Confluent's subject name strategies, see

<https://docs.confluent.io/platform/current/schema-registry/serdes-develop/index.html#subject-name-strategy>

From the **Subject compatibility mode** drop-down list, optionally change the default compatibility mode. The default is **Use Schema Registry defaults**.

For an explanation of the available modes, see <https://docs.confluent.io/platform/current/schema-registry/avro.html#compatibility-types>



When publishing messages to Confluent Schema Registry, the default subject compatibility mode for all newly created Control Table subjects will be *none*, regardless of the selected **Subject compatibility mode**.

Should you wish the selected **Subject compatibility mode** to apply to Control Tables as well, set the `setNonCompatibilityForControlTables` internal parameter to `false`.

Overriding the default settings

A transformation can be defined that overrides the topic and ordering key settings defined in the **General** tab.



Before you can define such a transformation, you first need to add a source endpoint to the task and select the tables you want to replicate.

To define a transformation:

1. Open the task you defined.
2. If you are defining a transformation for a single table, select one of the source tables. Otherwise, skip to Step 3.
3. Define a transformation that adds one of the following columns:



The columns listed below (prefixed with a \$) instruct Replicate to route the message to the desired topic and will not be included in the actual message itself.

- \$topic - To write messages to a specific topic.
- \$orderingkey - To create a custom ordering key.

For information on creating a transformation for a single table, see *Defining transformations for a single table/view* (page 1084).

For information on creating a global transformation rule, see *Starting the Global Transformation Rules wizard* (page 1107).

4. Define an expression for the new column that returns the following values:
 - For a \$topic column, the expression should return the topic name.
 - For a \$orderingkey column, the expression should return the ordering key contents. Note that the expression must return a non-empty value.

For information on creating expressions, see *Using the Expression Builder* (page 1137).

Setting advanced connection properties

In the **Advanced** tab, you can define advanced properties for the Google Cloud Pub/Sub target endpoint:

Performance tuning

If you experience performance issue, you can try adjusting these parameters to improve performance.

For each message batch:

- **Max number of messages:** The maximum number of messages to include in each batch sent to Google Cloud Pub/Sub.
- **Max batch size (bytes):** The maximum size of each batch to send.
- **Max batch latency (milliseconds):** The interval between publishing message batches.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary


You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

Data messages


This topic describes the structure and content of the data messages produced by the Google Cloud Pub/Sub target endpoint.

Data message		
Field	Type	Description
schema (name)	String	The name of the source database schema containing the replicated source table(s).
table (name)	String	The name of the source table.
headers	Structure	Information about the current record.
operation (Operation)	Enum	The operation type. Full Load - Initial load of the source data to the target topic(s) REFRESH – Insertion of a record to the target during Full Load CDC - Applies source table changes to the relevant topic INSERT – Insertion of new target record UPDATE – Update of an existing target record DELETE – Deletion of a target record
changeSequence (Change Sequence)	String	A monotonically increasing change sequencer that is common to all change tables of a task. Use this field to order the records in chronological order. Applicable to CDC operations only.
timestamp (Timestamp)	String	The original change UTC timestamp. Applicable to CDC operations only.

10 Adding and managing target endpoints

Field	Type	Description
transactionId (Transaction ID)	String	<p>The ID of the transaction that the change record belongs to.</p> <p>Use this field to gather all changes of a specific transaction.</p> <p>Applicable to CDC operations only.</p>
changeMask (Change Mask)	String	<p>Indicates which data columns were changed in the source table.</p> <p>The change mask is a string of hexadecimal digits, representing a bitmask of data columns in little-endian order. The bit position in the change mask is based on the ordinal of the column in the metadata message of that table.</p> <p>This means that if there are 10 data columns, they occupy bits 0 to 9 in the bitmask.</p> <p>If UPDATE mask is 0B hexadecimal, which is 1011 binary – it means that the columns at ordinals 1, 2 and 4 were changed.</p> <p>The following describes the bit semantics:</p> <ul style="list-style-type: none">• For INSERT records, all the inserted columns have the associated bits set.• For DELETE records, only primary-key (or unique index) columns have the associated bits set. This allows an applier to construct a DELETE statement without having to find the primary key fields from another source.• For UPDATE records, each column with a changed value will have the associated bit set. <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;"><p> <i>LOB columns are not included in the changeMask bit.</i></p></div>

10 Adding and managing target endpoints

Field	Type	Description
columnMask (Column Mask)	String	<p>Indicates which data columns are present in the message. Usually, this will include all of the table columns.</p> <div style="border: 1px solid gray; padding: 5px; margin: 10px 0;">  <i>When replicating from an Oracle source without full supplemental logging, some columns might not be present in the data, since they could not be replicated.</i> </div> <p>The column mask is a string of hexadecimal digits, representing a bitmask of data columns in little-endian order. The bit position in the column mask is based on the ordinal of the column in the metadata message for that table.</p> <p>This allows the applier to distinguish a null value that is the <i>actual</i> value of the column, from a null value that represents a column which could not be replicated from the source database.</p>
transactionEventCounter (Transaction Event Counter)	Long	<p>The sequence number of the current operation in the transaction.</p> <p>This can be used to determine the order of operations within a transaction.</p>
transactionLastEvent (Transaction Last Event)	Boolean	<p>"True" indicates that it is the final record in the transaction whereas "False" indicates that not all of the records have been processed.</p>
data	Structure	The data of the table record
{columns}		The column names and values in the current record.
beforeData	Structure	The data of the table record, before the change
{columns}		<p>The column names and values, before the change.</p> <p>Applicable to UPDATE operation.</p>

10.15 Using Google Cloud SQL for MySQL as a Target

This section describes how to set up and use a Google Cloud SQL for MySQL target endpoint in a replication task.

In this section:

- *Prerequisites (page 736)*
- *Limitations and considerations (page 737)*
- *Security requirements (page 737)*
- *Supported Data Types (page 737)*
- *Setting general connection properties (page 740)*
- *Setting advanced connection properties (page 741)*

Prerequisites

Before using Google Cloud SQL for MySQL as an endpoint in a Qlik Replicate task, make sure that the prerequisites outlined below have been met.

Driver setup

Installing the driver on Windows

Install MySQL ODBC Unicode Driver 64-bit version 8.0.32 on the Replicate machine.

Installing the driver on Linux

1. Stop the Replicate service:

```
./areplicate stop
```
2. Optionally, confirm that the service has stopped:

```
./areplicate status
```
3. Install MySQL ODBC Unicode Driver 64-bit version 8.0.32 on the Replicate machine.
4. Change the working directory to `<Replicate-Install-Dir>/bin`.
5. Copy the driver location to the `site_arep_login.sh` file:

```
echo "export LD_LIBRARY_PATH=\$LD_LIBRARY_PATH:/usr/lib64" >> site_arep_login.sh
```

This will add the driver to "LD_LIBRARY_PATH" and update the driver location in the **site_arep_login.sh** file.
6. Optionally, confirm that the driver location was copied:

```
cat site_arep_login.sh
```
7. Make sure that the `/etc/odbcinst.ini` file contains an entry for MySQL, as in the following example:

```
[MySQL ODBC 8.0 Unicode Driver]  
Driver = /usr/lib64/libmyodbc8w.so  
UsageCount = 1
```
8. Start the Replicate service:

```
./areplicate start
```
9. Optionally confirm that the service has started:

```
./areplicate status
```

General prerequisites

In addition to the client prerequisites detailed above, make sure that the following prerequisites have also been met:

- A Google Cloud SQL for MySQL account with the required *Security requirements (page 737)*.
- A Google Cloud SQL for MySQL database with the tables that you want to replicate should be accessible in your network.
The following Google Cloud SQL for MySQL editions are supported:
 - Google Cloud SQL for MySQL Community Edition
 - Google Cloud SQL for MySQL Standard Edition
 - Google Cloud SQL for MySQL Enterprise Edition
 - Google Cloud SQL for MySQL Cluster Carrier Grade Edition
- To replicate LOB columns when Google Cloud SQL for MySQL 8 is the target database, set the parameter `sort_buffer_size = 128M` in the MySQL database.

Replicating 4-byte UTF8 emojis

To support replication of 4-byte UTF8 emojis, the target schema character set must be set to `utf8mb4`.

Limitations and considerations

The following limitations apply:

- Due to the way MySQL operates, when loading data to a Google Cloud SQL for MySQL target during a Full Load task, duplicate key errors will not be reported to the logs.
- When updating a column's value to its existing value, a zero rows affected is returned from MySQL (unlike Oracle and Microsoft SQL Server that perform an update of one row). This generates an entry in the `attrep_apply_exceptions` control table and the following warning:
Some changes from the source database had no impact when applied to the target database. See `attrep_apply_exceptions` table for details.
- Due to an issue with MySQL 5.7 (which was resolved in version 8.0), Replicate cannot insert multibyte characters into TEXT columns when the Full Load uses CSV files. Consequently, if the target endpoint is MySQL 5.7 and a replicated table contains TEXT columns with UTF-8 multibyte characters, the target table might be empty.

Workaround:

In the **Advanced** tab, set the `loadUsingCSV` internal parameter to `FALSE`. Note that this might impact performance.

- If you need to replicate 4-byte emoji characters, the target schema character set must be set to `utf8mb4`.

Security requirements

You must provide Google Cloud SQL for MySQL account access to the Qlik Replicate user. This user must have read/write privileges in the Google Cloud SQL for MySQL database.

Additionally, for each target schema involved in the replication, the Qlik Replicate user must have the following permissions:

- ALTER
- CREATE
- CREATE TEMPORARY TABLES
- DELETE
- DROP
- INSERT
- SELECT
- UPDATE

Supported Data Types

The following table shows the Google Cloud SQL for MySQL database target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.

10 Adding and managing target endpoints

For information on how to view the data type that is mapped from the source, see the section for the source endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

Supported data types

Qlik Replicate Data Types	Google Cloud SQL for MySQL Data Types
BOOL	BOOL
BYTES	If length is => 1 and =< 8095, then: VARBINARY (Length) If length is => 8096 and =< 65535, then: BLOB If length is => 65536 and =< 16777215, then: MEDIUMBLOB If length is => 16777216 and =< 2147483647, then: LONGBLOB
DATE	DATE
TIME	TIME
DATETIME	If scale is => 0 and =< 6, then: DECIMAL (p,s) If scale is => 7 and =< 12, then: VARCHAR (37)
INT1	TINYINT
INT2	SMALLINT
INT4	INTEGER
INT8	BIGINT
NUMERIC	If scale is => 0 and =< 30, then: DECIMAL (p,s) If scale is => 31 and =< 100, then: VARCHAR (45)
REAL4	FLOAT
REAL8	DOUBLE

Qlik Replicate Data Types	Google Cloud SQL for MySQL Data Types
STRING	<p>If length is => 1 and =< 8095, then:</p> <p>VARCHAR (Length)</p> <p>If length is => 8096 and =< 65535, then:</p> <p>TEXT</p> <p>If length is => 65536 and =< 16777215, then:</p> <p>MEDIUMTEXT</p> <p>If length is => 16777216 and =< 2147483647, then:</p> <p>LONGTEXT</p>
UINT1	UNSIGNED TINYINT
UINT2	UNSIGNED SMALLINT
UINT4	UNSIGNED INTEGER
UINT8	UNSIGNED BIGINT
WSTRING	<p>If length is => 1 and =< 8095, then:</p> <p>VARCHAR (Length)</p> <p>If length is => 8096 and =< 65535, then:</p> <p>TEXT</p> <p>If length is => 65536 and =< 16777215, then:</p> <p>MEDIUMTEXT</p> <p>If length is => 16777216 and =< 2147483647, then:</p> <p>LONGTEXT</p>
BLOB	<p>When the Allow unlimited LOB size option is enabled:</p> <ul style="list-style-type: none"> • LONGBLOB <p>When the Limit LOB size to option is enabled:</p> <ul style="list-style-type: none"> • If the specified value <i>i</i> is 63 KB or smaller, then: BLOB • If the specified value is 64 KB or larger, then: LONGBLOB <p>For information on limiting LOB size, see <i>Target metadata (page 1166)</i>.</p>

Qlik Replicate Data Types	Google Cloud SQL for MySQL Data Types
NCLOB	<p>When the Allow unlimited LOB size option is enabled:</p> <ul style="list-style-type: none"> • TEXT <p>When the Limit LOB size to option is enabled:</p> <ul style="list-style-type: none"> • If the specified value is 63 KB or smaller, then: TEXT • If the specified value is 64 KB or larger, then: LONGTEXT <p>For information on limiting LOB size, see <i>Target metadata (page 1166)</i>.</p>
CLOB	<p>When the Allow unlimited LOB size option is enabled:</p> <ul style="list-style-type: none"> • TEXT CHARACTER SET UTF-16 <p>When the Limit LOB size to option is enabled:</p> <ul style="list-style-type: none"> • If the specified value is 63 KB or smaller, then: TEXT CHARACTER SET UTF-16 • If the specified value is 64 KB or larger, then: LONGTEXT CHARACTER SET UTF-16 <p>For information on limiting LOB size, see <i>Target metadata (page 1166)</i>.</p>

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a Google Cloud SQL for MySQL target endpoint to Qlik Replicate:

1. In the Qlik Replicate Console, click **Manage Endpoint Connections** to open the **Manage EndpointsConnections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Google Cloud SQL for MySQL database. This is optional.
4. Select **Target** as the database **role**.
5. From the **Type** drop-down list, select **Google Cloud SQL for MySQL**.
6. In the **Server** field, enter the host name or IP address of the computer on which the database is installed.



- *This information is case sensitive.*
- *To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**. If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box. To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.*

7. Optionally, change the default port (3306).
8. Type the Google Cloud SQL for MySQL authentication information (**User Name, Password**) for the authorized user for this database. If you do not know this information, see your MySQL database Administrator (DBA).
9. Select one of the following **Load source schemas into** options:
 - **The following database** - When this option is selected, all source schemas will be loaded into the selected database.
 - **Multiple endpoints** - When this option is selected, each of the source schemas will be loaded into its corresponding database.

Setting advanced connection properties

In the **Advanced** tab, you can set the following properties:

- **Max file size (KB)**: Select or type the maximum size (in KB) of a CSV file before it is loaded into the Google Cloud SQL for MySQL target database. The default value is 32000 KB.
- **Use parallel loading**: Select this option to improve performance when loading data into the Google Cloud SQL for MySQL target database.
 - **Use the following number of threads**: Specify how many threads to use to load the data into the Google Cloud SQL for MySQL target database. Note that setting a large number of threads may have an adverse effect on database performance since a separate connection is required for each thread.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.16 Using Google Cloud SQL for PostgreSQL as a target

This section describes how to set up and use a Google Cloud SQL for PostgreSQL target endpoint in a replication task.


In this section:

- *Prerequisites (page 744)*
- *Limitations (page 745)*
- *Security requirements (page 745)*
- *Google Cloud SQL for PostgreSQL database target data types (page 745)*
- *Setting general connection properties (page 746)*
- *Setting advanced connection properties (page 748)*

Prerequisites

This topic describes the client prerequisites when replicating to a Google Cloud SQL for PostgreSQL target.

Qlik Replicate Server for Windows

- **The PostgreSQL ODBC Driver:** PostgreSQL ODBC Driver [psqlodbc_13_01_0000-x64](#) or above must be installed on the Qlik Replicate machine.
- **PgAdmin:**
 1. Download and install  [PgAdmin 4 \(Windows\)](#) on the Qlik Replicate machine.
 2. Add the "psql.exe" path to the system PATH.
- Microsoft Visual Studio 2013 Redistributable (x86) - *not* (x64) - must be installed on the Qlik Replicate machine.

Qlik Replicate Server for Linux

Download PostgreSQL 13 ODBC driver, and client programs and libraries.

- **Example PostgreSQL ODBC file name:** postgresql13-odbc-13.02.0000-1PGDG.rhel8.x86_64.rpm
- **PostgreSQL client programs and libraries file name:** postgresql13-13.8-1PGDG.rhel<version>.x86_64.rpm

Then, open a shell prompt and do the following:

1. Stop the Replicate service:
`./areplicate stop`
2. Optionally, confirm that the service has stopped:
`./areplicate status`
3. Install PostgreSQL 13 client programs and libraries.
4. Install the ODBC driver for PostgreSQL 13.
5. Change the working directory to `<Replicate-Install-Dir>/bin`.
6. Copy the driver location to the `site_arep_login.sh` file:
`echo "export LD_LIBRARY_PATH=\$LD_LIBRARY_PATH:/usr/lib:/usr/lib64:/usr/pgsql-13/lib" >> site_arep_login.sh`
This will add the driver to "LD_LIBRARY_PATH" and update the driver location in the **site_arep_login.sh** file.
7. Optionally, confirm that the driver location was copied:
`cat site_arep_login.sh`
8. Makes sure that the `/etc/odbcinst.ini` file contains an entry for PostgreSQL, as in the following example:

```
[PostgreSQL Unicode(x64)]
Description = PostgreSQL ODBC driver
Driver = /usr/pgsql-13/lib/psqlodbc.so
Setup = /usr/pgsql-13/lib/psqlodbcw.so
UsageCount = 1
```
9. Start the Replicate service:
`./areplicate start`
10. Optionally, confirm that the service has started:
`./areplicate status`

Limitations

- UPDATE and DELETE operations on tables without a Primary Key/Unique Index that contain duplicate records is not supported. Any changes to these tables will affect all the records in the target.
- Tables with a LOB column in the key are not supported in Batch Optimized Apply mode. When a VARBINARY column is used as a source table key, a BYTEA column will be created in the target. This causes unpredictable behavior in Batch Optimized Apply mode. As a workaround, we suggest using the SQLite HEX function to convert VARBINARY to VARCHAR.

Security requirements

The user specified in the **General** tab when *Setting general connection properties (page 746)* must be a registered user in the Google Cloud SQL for PostgreSQL database.

In addition, the following privileges must be granted:

- Can login
- Create databases



If the target schema already exists in the database, the "Create databases" privilege is not required.

Google Cloud SQL for PostgreSQL database target data types

The Google Cloud SQL for PostgreSQL endpoint for Qlik Replicate supports most Google Cloud SQL for PostgreSQL database data types. The following table shows the Google Cloud SQL for PostgreSQL database target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types. Unsupported data types are listed below the table.

For information on how to view the data type that is mapped from the source, see the section for the source database you are using. For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

Qlik Replicate data types	Google Cloud SQL for PostgreSQL database data types
BOOL	BOOL
BYTES	BYTEA
DATE	DATE
TIME	TIME
DATETIME	If scale is => 0 and =< 6, then: TIMESTAMP If scale is => 7 and =< 12, then: VARCHAR (37)
INT1	SMALLINT

Qlik Replicate data types	Google Cloud SQL for PostgreSQL database data types
INT2	SMALLINT
INT4	INTEGER
INT8	BIGINT
NUMERIC	DECIMAL (P, S)
REAL4	FLOAT4
REAL8	FLOAT8
STRING	If length is 1 - 21845, then: VARCHAR (Length in Bytes = The STRING value multiplied by three) If length is 21846 - 2147483647, then: VARCHAR (65535)
UINT1	SMALLINT
UINT2	INTEGER
UINT4	BIGINT
UINT8	BIGINT
WSTRING	If length is 1 - 21845, then: VARCHAR (Length in Bytes = The WSTRING value multiplied by three) If length is 21846 - 2147483647, then: VARCHAR (65535)
BLOB	BYTEA
NCLOB	TEXT
CLOB	TEXT

Data types when Replicating from a PostgreSQL source

When replicating from a PostgreSQL source, the target table will be created with the same data types for all columns, apart from columns with user-defined or PostGIS data types. In such cases, the data type will be created as "character varying" in the target.

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a Google Cloud SQL for PostgreSQL target endpoint to Qlik Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Google Cloud SQL for PostgreSQL database. This is optional.
4. Select **Target** as the database **role**.
5. From the **Type** drop-down list, select **Google Cloud SQL for PostgreSQL**.
6. Type the **Server** name. This is the name or IP address or host name of the computer with the Google Cloud SQL for PostgreSQL database that you want to access.
7. Optionally, change the default port (5432).
8. Enter the Google Cloud SQL for PostgreSQL database authentication information (**User name**, **Password**) of an authorized Google Cloud SQL for PostgreSQL user. If you do not know this information, see your Google Cloud SQL for PostgreSQL database system manager.



Consider the following:

- *This information is case sensitive.*
- *To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**. If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box. To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.*



Make sure that the specified Google Cloud SQL for PostgreSQL database user has the correct access privileges.

9. Type the **Database name** or select one from the list of available endpoints. This is the name of the Google Cloud SQL for PostgreSQL database to which you are replicating data.

SSL authentication options

In the **Security** section, you can configure Replicate to connect to the Google Cloud SQL for PostgreSQL database using SSL.



The specified certificates are relevant only for bulk insert operations.

- **SSL Mode:** Select one of the following:
 - **Disable** - Connect with a surname and password only.
 - **Allow** - Establish an encrypted connection if requested by the server.

- **Prefer** - Establishes an encrypted connection if the server supports encrypted connections, falling back to an unencrypted connection if an encrypted connection cannot be established.
- **Require** - Establishes an encrypted connection if the server supports encrypted connections. The connection attempt fails if an encrypted connection cannot be established.
- **Verify CA** - Similar to **Required**, but also verifies the server Certificate Authority (CA) certificate against the configured CA certificates. The connection attempt fails if no valid matching CA certificates are found.
- **Verify Full** - Similar to **Verify CA**, but also performs host name identity verification by checking the host name the client (i.e. Replicate) uses for connecting to the server against the identity in the certificate that the server sends to the client. The client checks whether the host name that it uses for connecting matches the Common Name value in the server certificate. The connection fails if there is a mismatch.
- **Client certificate path** - The path to the client certificate requested by the server.
- **Secret key path** - The path to the client private key file in PEM format.
- **CA path** - The path to the Certificate Authority (CA) that issued the client certificate file in PEM format.
- **CRL path** - The path to the CRL certificate. This file contains certificates revoked by certificate authorities. If the server certificate appears in this list, the connection will fail.
- **SSL compression** - Select this option to compress the data before it is encrypted.

Setting advanced connection properties

In the **Advanced** tab, you can set the following properties:

You can set the following properties:

- **Max file size (KB)**: Select or type the maximum size (in KB) of a CSV file before the file is loaded into the Google Cloud SQL for PostgreSQL target database. The default value is 32000 KB.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

10 Adding and managing target endpoints

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.17 Using Google Cloud SQL for SQL Server as a target

This section describes how to set up and use a Cloud SQL for SQL Server database as a target in a replication task.



This endpoint is supported from Replicate May 2022 Service Release 02 only.



When configuring a task with a Google Cloud SQL for SQL Server target endpoint, note the following:

- If the [Apply Changes](#) task option is enabled and the change processing mode is set to [Batch optimized apply](#), it is recommended to enable the **Apply batched changes to multiple tables concurrently** option in the Change Processing Tuning (page 1181) tab.
- When the **Apply batched changes to multiple tables concurrently** option is selected, the option to set a Global Error Handling policy will not be available. Also, some of the task-specific error handling defaults will be different.
*For more information, see the **Apply batched changes to multiple tables concurrently** option in the Change Processing Tuning (page 1181) tab.*

In this section:

- [Supported editions \(page 751\)](#)
- [Prerequisites \(page 751\)](#)
- [Limitation and considerations \(page 752\)](#)
- [Permissions \(page 753\)](#)
- [Supported data types \(page 753\)](#)
- [Setting general connection properties \(page 754\)](#)
- [Setting advanced connection properties \(page 755\)](#)

Supported editions

The following Microsoft SQL Server editions are supported:

- Enterprise Edition
- Standard Edition
- Workgroup Edition
- Developer Edition
- SQL Server Web Edition

Prerequisites

Make sure the following prerequisites have been met:

Qlik Replicate for Windows

Install Microsoft ODBC Driver 18.1 for Windows on the Qlik Replicate Server machine.

Qlik Replicate for Linux

The procedure below assumes that you have installed a single default instance of Replicate on Linux (areplicate). If you have [installed multiple instances](#), replace areplicate with the name of the instance running the task with a Microsoft SQL Server source. If several instances are running such as task, the procedure needs to be repeated for each instance.

Install Microsoft ODBC Driver 18.1 for Linux on the Replicate machine.

Then, open a Unix shell and do the following:

1. Change the working directory to `<Replicate-Install-Dir>/bin`.
2. Stop the Replicate service:

```
./areplicate stop
```
3. Optionally, confirm that the service has stopped:

```
./areplicate status
```
4. Copy the driver location to the `site_arep_login.sh` file as follows:

```
echo "export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/microsoft/msodbcsql18/lib64/" >> site_arep_login.sh
```

This will add the driver to "LD_LIBRARY_PATH" and update the driver location in the **site_arep_login.sh** file.
5. Optionally, confirm that the driver location was copied:

```
cat site_arep_login.sh
```
6. Start the Replicate service:

```
./areplicate start
```
7. Optionally confirm that the service has started:

```
./areplicate status
```
8. Replicate requires the following ODBC library: `libmsodbcsql-18.1.so.1.1`
To check which library version is currently installed issue the following command:

```
ls /opt/microsoft/msodbcsql/lib64/
```

If the existing library has a different version number (e.g. `libmsodbcsql-18.0.so.1.1`), you need to create a symbolic link between the existing library and the required library.

To do this, issue the following commands:

```
cd /opt/microsoft/msodbcsql18/lib64/  
ln -s existing_library_name libmsodbcsql-18.1.so.1.1
```

where *existing_library_name* is the name of the currently installed library (e.g. libmsodbcsql-18.0.so.1.1).

TLS 1.2 prerequisites

If your environment matches the following:

- Microsoft SQL Server 2014
- Microsoft ODBC Driver 18 installed on Red Hat 8.1 or later

Then, in order to work with TLS 1.2, you must install a specific Service Pack on the Windows machine where Microsoft SQL Server is installed.

For more information, see <https://support.microsoft.com/en-us/help/3135244/tls-1-2-support-for-microsoft-sql-server>.

Limitation and considerations

When using a Microsoft SQL Server database as a target in a Replicate task, The following imitations apply:

- When **Use BCP for loading tables** is selected in the endpoint settings' **Advanced** tab (the default), the following limitations apply:
 - Full-sized LOB columns are not supported in **Batch optimized apply** change processing mode. You can work around this limitation by [limiting LOB column size in the task settings](#), clearing the **Use BCP for loading tables** option, or switching to **Transactional apply** mode.
 - If following large object types are empty in the source database (0 bytes), they will not be empty after Full Load:
 - **BLOB** will be written as 0x0000 (2 bytes)
 - **TEXT** (VARCHAR(max)) will be written as 0X00 (1 byte)
 - **NTEXT** (NVARCHAR(max)) will be written as 0x0000 (2 bytes)



If these types are empty during Change Processing, they will be correctly written to the target as empty (0 bytes).

- Triggers are not executed.

For more information on the **Use BCP for loading tables** mode, see *Setting advanced connection properties (page 755)*.

- When both the **Batch optimized apply** and **Limit LOB size to** options are enabled, BLOB columns on the target will be truncated to a quarter of the size specified in the [Limit LOB size to](#) field.
- If you need to replicate 4-byte emoji characters, use a transformation to convert the data type from WSTRING(n) to WSTRING (n*2).

Permissions

The user defined in the endpoint connection settings must have at least the `db_owner` user role on the target Microsoft SQL Server database.

Supported data types

The Google Cloud SQL for SQL Server target for Qlik Replicate supports most Microsoft SQL Server data types. The following table shows the Microsoft SQL Server target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source database you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.



When replicating to Microsoft SQL Server 2019, UTF-8 data can be transferred to VARCHAR, CHAR, and CLOB columns when the database collation is set to %collation%_UTF8 (for example, **Latin1_General_100_CI_AS_SC_UTF8**). When working with an earlier Microsoft SQL Server version or a different collation, use a transformation (global or table-specific) to transform the column data type to WSTRING.

See also *Using the Transform tab (page 1086)* and *Defining global rules (page 1107)*.

Supported target data types

Qlik Replicate Data Types	Microsoft SQL Server Data Types
BOOLEAN	TINYINT
BYTES	VARBINARY(length)
DATE	DATE
TIME	TIME(0)
DATETIME	DATETIME2(scale)
INT1	SMALLINT
INT2	SMALLINT
INT4	INT
INT8	BIGINT
NUMERIC	NUMERIC(p,s)
REAL4	REAL
REAL8	FLOAT

Qlik Replicate Data Types	Microsoft SQL Server Data Types
STRING	<p>If column is date or time then:</p> <p>DATETIME2</p> <p>If the column is <i>not</i> a date or time:</p> <p>VARCHAR(length)</p>
UINT1	INT2
UINT2	INT4
UINT4	INT8
UINT8	NUMERIC(20)
WSTRING	NVARCHAR(length)
BLOB	<p>VARBINARY(max)</p> <p>IMAGE</p> <p>To use this data type with Qlik Replicate, you must enable the use of BLOBs for a specific task.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p>
CLOB	<p>VARCHAR(max)</p> <p>TEXT</p> <p>To use this data type with Qlik Replicate, you must enable the use of CLOBs for a specific task.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p>
NCLOB	<p>NVARCHAR(max)</p> <p>NTEXT</p> <p>To use this data type with Qlik Replicate, you must enable the use of NCLOBs for a specific task.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p>

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a **Google Cloud SQL for SQL Server** target endpoint to Qlik Replicate:

1. In the Qlik Replicate Console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box and then click **New Endpoint Connection**. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Microsoft SQL Server database. This is optional.
4. Select **Target** as the database **role**.
5. Select **Google Cloud SQL for SQL Server** as the database **Type**.
6. In the **Server name** field, specify the host name or IP address of the computer with the Microsoft SQL Server instance containing the target database.



*To override the default port, add the port to the server name, separated by a comma. For example, if the server name is `myserver.company.local` and the port is `3333`, then the server name should be written like this:
`myserver.company.local,3333`*

7. In the **User name** field, enter the user name of an authorized database user.
8. In the **Password** field, enter the password of the specified user.
9. Type the **Database name** or click **Browse** and select one from the list of available databases. This is the name of the database to where you are replicating the data.

Setting advanced connection properties

In the **Advanced** tab, you can set the following properties:

- **Use BCP for loading tables:** Select this to transfer data for full-load operations using BCP.



*When the target table contains an identity column that does not exist in the source table, you must disable the **Use BCP for loading tables** option.*

- **BCP packet size:** The maximum size of the packets (in bytes) used to transfer data using BCP.
- **File group for Qlik Replicate internal tables:** Optionally, specify a filegroup for the Qlik Replicate internal tables. When the replication task starts, all of the internal Qlik Replicate [control tables](#) will be created in the specified filegroup.

The following is an example of a command for creating a filegroup:

```
ALTER database replicate
ADD FILEGROUP Test1FG1;
GO
ALTER database replicate
ADD FILE
(
NAME = test1dat5,
FILENAME = 'C:\temp\DATA\t1dat5.ndf',
SIZE = 5MB,
MAXSIZE = 100MB,
```

```
FILEGROWTH = 5MB
)
TO FILEGROUP Test1FG1;
GO
```

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.18 Using Google Cloud Storage as a target

This chapter describes how to set up and use Google Cloud Storage as a target in a replication task.

In this section:

- *Prerequisites (page 758)*
- *Google Cloud storage target overview (page 758)*
- *Limitations and considerations (page 758)*
- *Change data partitioning (page 759)*
- *Data types (page 759)*
- *Setting general connection properties (page 761)*
- *Setting advanced connection properties (page 765)*
- *Generating reference files (page 767)*

Prerequisites

Before you can use Google Cloud Storage as a target endpoint in a Replicate task, the following prerequisites must be met:

- **Permissions:** The JSON credentials specified in the Google Cloud Storage endpoint's **Google Cloud Storage** settings must be for an account that has read and write access to the specified bucket and folder.

Google Cloud storage target overview

When using Google Cloud Storage as a target in a Replicate task, both the Full Load and Change Processing data are written to data files. Depending on the endpoint settings, the data file format can be CSV, JSON or Parquet. While the explanations in this section relate to CSV files, the same is true for JSON and Parquet files.

Full Load files are named using incremental counters e.g. LOAD00001.csv, LOAD 00002.csv, etc. whereas Change Processing files are named using timestamps e.g. 20141029-1134010000.csv.



*When the [Create metadata files in the target folder](#) option is enabled, a corresponding metadata file is created using the same naming format, but with a **.dfm** extension.*

For each source table, a folder is created in the specified Google Cloud Storage target folder. The data files are created on the Replicate Server machine and are then uploaded to the specified Google Cloud Storage target folder once the [File Attributes](#) (Full Load) and [Change Processing](#) upload conditions have been met.

DDL handling

When a DDL change is captured, Replicate will close the data file and also create a DFM file if the **Create metadata files in the target folder** option is enabled. When the next batch of changes arrives, Replicate will create a new data file containing the changes. Note that the DFM file created for the new data file will match the new table structure.

Limitations and considerations

The following limitations apply to the Google Cloud Storage target endpoint:

- The following DDLs are supported only: Truncate table, Drop table, Create table, Add Column, Rename Column, Drop Column, and Convert Data Type.
- Full LOB Mode is not supported
- UPDATE and DELETE statements are not supported in Apply Changes replication mode
- Batch optimized apply mode is not supported
- Target lookup is not supported
- The <target folder> parameter cannot include special characters
- Limitations when using Parquet file format:
 - When the [DDL Handling Policy](#) for source tables is set to **Ignore ALTER**, the RENAME COLUMN DDL (on a Primary Key column) is not supported.
 - LOB columns larger than 1 MB are not supported.

- Database names, schema names, or table names containing slash (/) or backslash (\) characters are not supported.
- The following Control Tables are not supported as they require UPDATE/DELETE operations, which are not supported:
 - **Replication Status** (requires UPDATE).
Name on target: `attrep_status`
 - **Suspended Tables** (requires DELETE).
Name on target: `attrep_suspended_tables`
For more information on Control Tables, see *Control tables (page 1169)*.

Change data partitioning

When replicating to a Google Cloud Storage target, for each of the source tables, a directory is created under the specified target directory. When *Change Data Partitioning (page 1177)* is enabled, an additional sub-directory is created under the corresponding table directory. The data and metadata (when the metadata option is enabled) files are located in the partition subdirectory, as in the following example:

```
{Target Directory}
  {Table_1}
    {Partition_1}
      Data files
      DFM files
    {Partition_2}
      Data files
      DFM files
    {Partition_3}
      Data files
      DFM files
  {Table_2}
    {Partition_1}
      Data files
      DFM files
    {Partition_2}
      Data files
      DFM files
    {Partition_3}
      Data files
      DFM files
```

Information about the partitions is written to the **attrep_cdc_partitions** Control Table. For information about this table, see *Change data partitions (page 1301)*.

Data types

The following table shows the default mapping from Qlik Replicate data types to Google Cloud Storage target data types. Note that the data type mapping is only relevant if the [Create metadata files in the target folder](#) option is enabled.

For information on source data type mappings, see the section for the source endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

10 Adding and managing target endpoints

Qlik Replicate and Google Cloud Storage data types

Qlik Replicatedata types	Google Cloud Storage Target data types
DATE	DATE
TIME	TIME
DATETIME	DATETIME
BYTES	BYTES (length)
BLOB	BLOB
REAL4	REAL4 (7)
REAL8	REAL8 (14)
INT1	INT1 (3)
INT2	INT2 (5)
INT4	INT4 (10)
INT8	INT8 (19)
UINT1	UINT1 (3)
UINT2	UINT2 (5)
UINT4	UINT4 (10)
UINT8	UINT8 (20)
NUMERIC	NUMERIC (p,s)
STRING	STRING (Length)
WSTRING	STRING (Length)
CLOB	CLOB
NCLOB	NCLOB
BOOLEAN	BOOLEAN (1)

Mapping from Qlik Replicate data types to Parquet

When Parquet is set as the message format, due to the limited number of data types supported by Parquet, the data type mappings will be as shown in the table below.

Parquet data type mappings

Replicate Data Type	Parquet Primitive Type	Logical Type
BOOLEAN	BOOLEAN	
INT1	INT32	INT(8, true)
INT2	INT32	INT(16, true)
INT4	INT32	

Replicate Data Type	Parquet Primitive Type	Logical Type
INT8	INT64	
UINT1	INT32	INT(8, false)
UINT2	INT32	INT(16, false)
UINT4	INT64	
UINT8	INT64	INT(64, false)
REAL4	FLOAT	
REAL8	DOUBLE	
NUMERIC	FIXED_LEN_BYTE_ARRAY (16)	DECIMAL (precision, scale)
STRING	BYTE_ARRAY	STRING
WSTRING	BYTE_ARRAY	STRING
BYTES	BYTE_ARRAY	
BLOB	BYTE_ARRAY	
CLOB	BYTE_ARRAY	STRING
NCLOB	BYTE_ARRAY	STRING
DATE	INT32	DATE
TIME	INT32	TIME(UTC=true, unit=MILLIS)
DATETIME	INT64	TIMESTAMP (UTC=true, unit=MICROS)

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.





To add a Google Cloud Storage target endpoint to Qlik Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.
2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the endpoint being used.
3. Optionally, in the **Description** field, type a description that helps to identify the endpoint.
4. Select **Target** as the endpoint **role**.

10 Adding and managing target endpoints

5. Select **Google Cloud Storage** as the endpoint **Type**.
6. Configure the remaining settings in the **General** tab as described in the table below.

Google Cloud Storage option descriptions

Option	Description
JSON credentials	The JSON credentials for the service account key with read and write access to the Google Cloud Storage bucket.
Bucket name	The Google Cloud Storage bucket.
Target folder	Where to create the data files in the specified bucket.
File Attributes	<p>Delimiters can be standard characters or a hexadecimal (hex) value. Note that the "0x" prefix must be used to denote a hexadecimal delimiter (e.g. 0x01 = SOH). In the Field delimiter, Record delimiter and Null value fields, the delimiter can consist of concatenated hex values (e.g. 0x0102 = SOHSTX), whereas in the Quote character and Escape character fields, it can only be a single hex value.</p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p> <i>The hexadecimal number 0x00 is not supported (i.e. only 0x01-0xFF are supported).</i></p> </div>
Format	<p>You can choose to create the target files in CSV, JSON or Parquet format.</p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p> <i>Parquet format is supported from Replicate May 2022 Service Release 02 only.</i></p> </div> <p>In a JSON file, each record is represented by a single line, as in the following example:</p> <pre>{ "book_id": 123, "title": "Alice in Wonderland", "price": 6.99, "is_hardcover": false } { "book_id": 456, "title": "Winnie the Pooh", "price": 6.49, "is_hardcover": true } { "book_id": 789, "title": "The Cat in the Hat", "price": 7.23, "is_hardcover": true }</pre> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p> <i>Changing the format (for example, from CSV to JSON or from JSON to CSV) while the task is in a stopped state and then resuming the task, is not supported.</i></p> </div> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p> <i>If you choose JSON format, the following fields will be hidden as they are only relevant to CSV format: Field delimiter, Record delimiter, Null value, Quote character, Escape character, and Add metadata header.</i></p> </div> <p>For information about data type mappings when using Parquet format and limitations, see <i>Mapping from Qlik Replicate data types to Parquet (page 760)</i> and <i>Limitations and considerations (page 758)</i>.</p>

Option	Description
Field delimiter	<p>The delimiter that will be used to separate fields (columns) in the target files. The default is a comma.</p> <p>Example using a comma as a delimiter:</p> <pre>"mike","male"</pre>
Record delimiter	<p>The delimiter that will be used to separate records (rows) in the target files. The default is a newline (\n).</p> <p>Example:</p> <pre>"mike","male"\n "sara","female"\n</pre>
Null value	<p>The string that will be used to indicate a null value in the target files.</p> <p>Example (where \n is the record delimiter and @ is the null value):</p> <pre>"mike","male",295678\n "sara","female",@\n</pre>
Quote character	<p>The character that will be used at the beginning and end of a text column. The default is the double-quote character ("). When a column that contains column delimiters is enclosed in double-quotes, the column delimiter characters are interpreted as actual data, and not as column delimiters.</p> <p>Example (where a @ is the quote character):</p> <pre>@mike@,@male@</pre>
Quote escape character	<p>The character used to escape a quote character in the actual data. The default is the double-quote character (").</p> <p>Example (where " is the quote character and \ is the escape character):</p> <pre>1955,"old, \"rare\", Chevrolet","\$1000"</pre>

10 Adding and managing target endpoints

Option	Description
Add metadata header	<p>When the target storage format is set to Text, you can optionally add a header row to the data files. The header row can contain the source column names and/or the intermediate (i.e. Replicate) data types.</p> <p>Example of a target file with a header row when both With column names and With data types are selected:</p> <p>Position:DECIMAL(38,0),Color:VARCHAR(10)</p> <p>1,"BLUE"</p> <p>2,"BROWN"</p> <p>3,"RED"</p> <p>...</p>
Maximum file size	<p>The maximum size a file can reach before it is closed (and optionally compressed). This value applies both to data files and to Reference Files.</p> <p>For information on generating reference files, see <i>Setting advanced connection properties (page 765)</i>.</p>
Compress files using	<p>Choose one of the compression options to compress the target files or NONE (the default) to leave them uncompressed. Note that the available compressions options are determined by the selected file format.</p>

Change Processing option descriptions

Option	Description
Apply/Store changes when file size reaches	Specify the maximum size of Change Data to accumulate before uploading the file to Google Cloud Storage .
Apply/Store changes when Elapsed time reaches	Specify the maximum time to wait before applying the changes.

Metadata files option descriptions

Option	Description
Create metadata files in the target folder	<p>When this option is selected, for each data file, a matching metadata file with a .dfm extension will be created under the specified target folder. The metadata files (which are in standard JSON format) provide additional information about the task/data such as the source endpoint type, the source table name, the number of records in the data file, and so on.</p> <p>For a full description of the metadata file as well as possible uses, see <i>Metadata file description (page 1310)</i>.</p>

7. To determine if the connection information you entered is correct, click **Test Connection**. If the connection test is successful, click **Save**.

Setting advanced connection properties

In the **Advanced** tab, you can set post-processing actions and enable the creation of reference files. These options are described in detail below.

Post upload processing

Run command after upload

Select **Run command after upload** if you wish to process the final target files using a custom command. The command will be run whenever a data file is created.



*If the **Generate Reference Files** option is also selected, a row (specifying the file's location) will be added to the Reference File, but only if the command completes successfully.*

Command name

The location of the command e.g. `C:\utils\move.exe`.

Working directory

The directory where you want the command to run.

Parameters

Specify any parameters that need to be passed to the command during runtime. You can use the following built-in parameters:

`${FILENAME}` - The full path to the CSV file containing the full load or CDC data.

`${METADATA_FILENAME}` - The full path to the DFM file containing the metadata.

For information on creating metadata files, see *Setting general connection properties (page 761)*.



If the CSV/DFM file paths contain spaces, you must enclose these parameters with quotation marks (e.g. `"${FILENAME}"`).

Standard post command exit codes

The post-processing command must return a proper exit code. You can either use the standard exit code values described below or set a custom exit code value as described in **Setting post command exit codes with an internal parameter** below.

- **0** - Success
- **1** - Recoverable error. The task will recover from the point of failure according to the settings in the *Environmental errors (page 1187)* tab.
- **2** - Table error. If a table error occurs, Replicate will handle the error according to the settings in the

Table Errors (page 1191) tab.

- **3**(or any other value e.g. -100) - Fatal error. The task will fail and not attempt recovery.

Setting post command exit codes with an internal parameter

You can use internal parameters to set exit codes with custom values. This is especially useful if your application already uses the standard exit code values.

See **Standard post command exit codes** above for a description of the exit codes.

- `successExitCode`
- `recoverableErrorExitCode`
- `tableErrorExitCode`
- `fatalErrorExitCode`

For instructions on setting internal parameters, see *Internal parameters (page 767)*.

Generate reference files

Select the **Generate Reference Files** option to generate a Reference File (on Replicate Server) containing the full path to the Apply Changes data files.



The reference file only points to the location of the Change Processing files, and not the Full Load files.

Reference File(s) folder

The folder on the Replicate machine in which the Reference File will be created.

Example:

`c:\temp\`

Use proxy server

Select the **Use proxy server** option to access Google Cloud Storage via a proxy server.

Host name

The host name of the proxy server.

Port

The port via which to access the proxy server.

User name

The user name for accessing the proxy server.

Password

The password for accessing the proxy server.

Scheme

Select which protocol to use to access the server (HTTP or HTTPS).

SSL CA Path

The location of the CA file on the Replicate Server machine when **HTTPS** is the selected **Scheme**.

Internal parameters

Internal parameters are parameters that are not exposed in the UI and should only be used if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your setting by clicking the **View Setting Summary** link. This is useful if you need to send a summary of your setting to Qlik Support.

Generating reference files

In the **Advanced** tab of the Google Cloud Storage target endpoint, you can enable the **Generate a reference file** option. The Reference File contains a list of the Change File locations and is therefore only relevant if the task's Apply Changes or Store Changes options are enabled.

The format of the reference file name is as follows:

```
<google_cloud_storage_target_endpoint_name><counter>.csv|json
```

Example:

```
MyGoogleCloudStorage00000001.csv
```



The counter suffix increases incrementally each time a new Reference File is generated (i.e. which occurs when the file reaches the maximum size defined [in the General tab](#)). Once a new Reference File has been generated, you can delete the old reference file(s) if required.

10 Adding and managing target endpoints

Whenever an Change Processing data file is created, a new row is added to the Reference File in the following format:

```
<Source_Table_Name>,<Data_Lake_Store_name>/<path>/<file_name>
```

Example:

```
employees,mydatalakestore/new/files/my.company/20170611-120144192.csv
```

Note that if the **Run command after upload** option in the **Advanced** tab is also enabled, the Reference File will be generated *after* the post-processing completes.

10.19 Using Google Dataproc as a target

This section describes how to set up and use Google Dataproc as the target endpoint in a replication task.

In this section:

- *Prerequisites (page 770)*
- *Limitations (page 770)*
- *Change data partitioning on Google Dataproc (page 771)*
- *Google Dataproc endpoint target data types (page 771)*
- *Setting general connection properties (page 772)*
- *Setting advanced connection properties (page 773)*

Prerequisites

Before you begin to work with Google Dataproc as a target in Qlik Replicate, make sure that the following prerequisites have been met:

- **General:**
 - The storage location must be accessible from the Qlik Replicate machine.
 - The user specified in the Google Dataproc target endpoint's [Hive access settings](#) must have access to HiveServer2.
- **ODBC Driver (Windows and Linux):** Install Cloudera ODBC driver for Apache Hive 2.6.9 or later, or any other Google Dataproc compatible driver on the Replicate Server machine.
- **Bucket and folder permissions:** The **JSON Credentials** specified in the Google Dataproc endpoint's **Google Cloud Storage** settings must be for an account that has read and write access to the specified bucket and target folder.
- **Hive table permissions:** Replicate requires permissions to perform the following operations on Hive tables: CREATE, DROP, DESCRIBE, and ALTER table. ALTER table may also include RENAME table, ADD/RENAME column, and CREATE/DROP partitions on tables.

Limitations

The following limitations apply:

- UPDATE/DELETE DMLs are not supported during change processing. If an UPDATE/DELETE DML was captured on the source, it will be ignored on the target and a warning will be written to the log. If the **Store Changes** option is enabled in the task settings, these records will be written to the Change Table.
- Limited LOB support only.
- Dropping columns and changing column data types or the data type length is not supported and will suspend the table in all cases *except* for data types that are mapped to STRING. Changes to the data type length of a data type mapped to STRING (e.g. VARCHAR(50) to VARCHAR(100)) will simply be ignored.
- Writing to "special" tables such as tables with partitions, buckets, or skews is not supported.
- The following Control Tables are not supported as they require UPDATE/DELETE operations, which are not supported:
 - **Replication Status** (requires UPDATE).
Name on target: `attrep_status`
 - **Suspended Tables** (requires DELETE).
Name on target: `attrep_suspended_tables`
For more information on Control Tables, see *Control tables (page 1169)*.
- Table and column names can only contain ASCII characters.
- Column names cannot contain the '?' symbol. If needed, the '?' symbol can be replaced with a valid ASCII character [using a transformation](#).

- Google Cloud Storage limitations:
 - Append operation is not supported
- Proxy:
 - Does not affect the ODBC (Hive) connection (i.e. it affects the storage connection only).

Change data partitioning on Google Dataproc

When *Change Data Partitioning* (page 1177) is enabled, the Replicate Change Tables in Hive are partitioned by the `partition_name` column. Data files are uploaded to Google Cloud Storage, according to the [maximum size and time definition](#), and then stored in a directory under the Change Table directory. Whenever the specified partition timeframe ends, a partition is created in Hive, pointing to Google Cloud Storage.

Information about the partitions is written to the [attrep_cdc_partitions](#) Control Table.

Google Dataproc endpoint target data types

The following table shows the Google Dataproc endpoint target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).

Qlik Replicate and Google Dataproc data types

Qlik Replicatedata types	Google Dataproc data types
BOOL	BOOLEAN
BYTES	STRING
TIME	STRING
DATETIME	TIMESTAMP
DATE	DATE
INT1	TINYINT
INT2	SMALLINT
INT4	INT
INT8	BIGINT
NUMERIC	DECIMAL (p,s)
REAL4	FLOAT
REAL8	DOUBLE
STRING	VARCHAR (Length)
UINT1	SMALLINT
UINT2	INT

Qlik Replicatedata types	Google Dataproc data types
UINT4	BIGINT
UINT8	DECIMAL (20,0)
WSTRING	VARCHAR (Length)
BLOB	STRING
NCLOB	STRING
CLOB	STRING

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a Google Dataproc target endpoint to Qlik Replicate:

1. In the Qlik Replicate console, click **Manage Endpoint Connections** to open the **Manage Endpoint Connections** dialog box.
For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.
2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the endpoint being used.
3. In the **Description** field, type a description that helps to identify the Google Dataproc endpoint. This is optional.
4. Select **Google Dataproc** as the endpoint **Type**.
5. Configure the remaining settings in the **General** tab as described in the table below.

Google Cloud Storage option descriptions

Option	Description
JSON credentials	The JSON credentials for the service account key with read and write access to the Google Cloud Storage bucket.
Bucket name	The Google Cloud Storage bucket.
Target folder	Where to create the data files in the specified bucket.


Hive access option descriptions

Option	Description
Host	The host name of the Hive server.
Port	The port of the Hive server. The default is 10,000.
Authentication type	Select either User name or User name and Password .
Username	The user name for accessing the Hive server.


Password	The password for accessing the Hive server.
Database	The name of the Hive target database.

Setting advanced connection properties


The table below describes the settings in the **Advanced** tab.

Setting	Description
File Format	Expand this section to specify or view the file format settings.
Target storage format	Select either Text or Sequence . <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;">  <i>When using Text as the target storage format, new lines as part of the data values are not supported (due to an Apache Hadoop limitation). Best practice is to use Sequence as the target storage format.</i> </div>
Field delimiter	The delimiter that will be used to separate fields in the target file. The default is \001. Note that field delimiters should be ASCII characters unless they are non-printable control characters, in which case they should be in Octal format (e.g. \247).
Null value	The value that will be used to indicate a null value in the target file. Example (where @ is the null value): mike,male,295678 sara,female,@
Escape character	The escape character is used to escape the field delimiter character. When a field delimiter is escaped, it is interpreted as actual data, and not as a field delimiter. Example (where \ is the escape character and a comma is the field delimiter): sunroof\,power-steering

10 Adding and managing target endpoints

Setting	Description
Add metadata header	<p>When the target storage format is set to Text, you can optionally add a header row to the data files. The header row can contain the source column names and/or the intermediate (i.e. Replicate) data types.</p> <p>Example of a target file with a header row when both With column names and With data types are selected:</p> <p>Position:DECIMAL(38,0),Color:VARCHAR(10)</p> <p>1,"BLUE"</p> <p>2,"BROWN"</p> <p>3,"RED"</p> <p>...</p>
File Attributes	Expand this section to specify or view the file attributes.
Maximum file size	Specify the maximum file size of each target file. When the data reaches the maximum size, the file will be closed and written to the specified target folder.
Compress files using	Select the compression method to use on Google Cloud Storage.
Change Processing	Expand this section to specify or view change processing settings.
Apply/Store changes when:	
File size reaches	Specify the minimum size of the data required to create a file in idle state.
Elapsed time reaches	Specify the maximum time to wait before applying the changes in idle state.
<div style="border: 1px solid #ccc; padding: 10px; margin-bottom: 10px;">  <p><i>To facilitate rapid delivery of DDL messages, files are uploaded immediately, regardless of the specified File size reaches or Elapsed time reaches values.</i></p> </div>	
Proxy Server	Expand this section to specify or view proxy settings.
Use proxy server	Select this option to access Google Dataproc via a proxy server.

10 Adding and managing target endpoints

Setting	Description
Host name	The host name of the proxy server.
Port	The port via which to access the proxy server.
User name	The user name for accessing the proxy server.
Password	The password for accessing the proxy server.
Scheme	Select which protocol to use to access the server (HTTP or HTTPS). <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <i>The proxy configuration applies to the Google Cloud Storage, but not to the ODBC connection.</i></div>

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.20 Using Hadoop as a target

This section describes how to set up and use Hadoop as the target endpoint in a replication task.

In this section:

- *Prerequisites (page 777)*
- *Limitations (page 777)*
- *Change Data Partitioning on Hadoop (page 778)*
- *Security requirements (page 779)*
- *Hadoop endpoint target data types (page 779)*
- *Setting general connection properties (page 780)*
- *Setting advanced connection properties (page 785)*
- *Using Kerberos authentication (page 792)*

Prerequisites

Before you begin to work with a Hadoop cluster as a target in Qlik Replicate, make sure that the following prerequisites have been met:

- **General:**

- The Hadoop WebHDFS must be accessible from the Qlik Replicate machine.
- The Hadoop Data Nodes must be accessible from the Qlik Replicate machine.
- The Hadoop WebHDFS service must be running.

- **ODBC Access:**

When accessing Hive using ODBC, the following ODBC drivers are supported:

- **Hortonworks:** Cloudera ODBC Driver for Apache Hive 2.06.09.1009 only.
- **Cloudera:** Cloudera ODBC Driver for Apache Hive 2.06.09.1009 only.



Cloudera ODBC drivers 2.5.20 or later do not support the [Snappy](#) compression method.

- **SSL:** Before you can use SSL, you first need to perform the following tasks:

- Configure each NameNode and each DataNode with an SSL certificate (issued by the same CA).
- Place the CA certificate on the Replicate Server machine. The certificate should be a base64-encoded PEM (OpenSSL) file.

- **Permissions:** The user specified in the Hadoop target settings must have write permission for the specified HDFS target directory.

Prerequisites for using the Cloudera Distribution as a Hadoop target

If you are replicating to a Cloudera Hadoop Distribution and you want to use **Snappy** compression and/or set a **File Format** that is **not** Text, you first need to install Cloudera's Hive ODBC driver on the Replicate Server machine. Then configure the Hadoop target endpoint to access Hive using ODBC. For more information on this setting, see *Setting general connection properties (page 780)*

See also *Prerequisites for using a Linux ODBC driver (page 777)*.

Prerequisites for using a Linux ODBC driver

To use a Linux ODBC driver, make sure to:

- Install the latest 64-bit ODBC driver for your Hadoop distribution on the Replicate Server machine.
- *After* the driver is installed: Edit the <distribution>.hiveodbc.ini file as follows:
 - `DriverManagerEncoding=UTF-16`
 - `ODBCInstLib=libodbcinst.so`

See also *Setting general connection properties (page 780)*.

Limitations

The following limitations apply:

- UPDATE/DELETE DMLs are not supported during change processing. If an UPDATE/DELETE DML was captured on the source, it will be ignored on the target and a warning will be written to the log. If the **Store Changes** option is enabled in the task settings, these records will be written to the Change Table.
- Limited LOB support only.
- Dropping columns and changing column data types or the data type length is not supported and will suspend the table in all cases *except* for data types that are mapped to STRING. Changes to the data type length of a data type mapped to STRING (e.g. VARCHAR(50) to VARCHAR(100)) will simply be ignored.
- The Replicate Hadoop target endpoint does not support the creation of skews, buckets or partitions in the target tables. Tables with such properties must already exist on the target *before* the task starts. For more information on preparing such tables in Hive, see [Support for Partitions, Buckets and Skews](#). When loading data into existing skewed, bucketed or partitioned target tables, the following limitations apply:
 - The **DROP and CREATE table** and **ARCHIVE and CREATE table** options in the task settings' *Full Load Settings (page 1171)* tab should not be selected (as the new table will be created without the special properties).
 - The Apply Changes replication option is not supported.
- The following Control Tables are not supported as they require UPDATE/DELETE operations, which are not supported:
 - **Replication Status** (requires UPDATE).
Name on target: `attrep_status`
 - **Suspended Tables** (requires DELETE).
Name on target: `attrep_suspended_tables`
For more information on Control Tables, see *Control tables (page 1169)*.
- Table and column names can only contain ASCII characters.
- Column names cannot contain the '?' symbol. If needed, the '?' symbol can be replaced with a valid ASCII character [using a transformation](#).

Change Data Partitioning on Hadoop

When *Change Data Partitioning (page 1177)* is enabled, the Replicate Change Tables in Hive are partitioned by the `partition_name` column. Data files are uploaded to HDFS, according to the [maximum size and time definition](#), and then stored in a directory under the Change Table directory. Whenever the specified partition timeframe ends, a partition is created in Hive, pointing to the HDFS directory.

Information about the partitions is written to the [attrep_cdc_partitions](#) Control Table.

Prerequisites

The prerequisites for using Change Data Partitioning with a Hadoop target endpoint are as follows:

- The target file format must be set to Text or Sequence
- Hive access must be set to ODBC

Security requirements


- The Hadoop NameNode (and data nodes when using WebHDFS) must be accessible from the Qlik Replicate machine and the user specified in the Hadoop target settings must have write permission for the specified HDFS target directory.
- **Hive table permissions:** Replicate requires permissions to perform the following operations on Hive tables: INSERT from SELECT, CREATE, DROP, DESCRIBE, and ALTER table. ALTER table may also include RENAME table, ADD/RENAME column, and CREATE/DROP partitions on tables.

Hadoop endpoint target data types

The following table shows the Hadoop endpoint target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).

Qlik Replicatedata types	Hadoop data types
BOOL	BOOLEAN
BYTES	STRING
TIME	STRING
DATETIME	TIMESTAMP
DATE	DATE
<div style="border: 1px solid #ccc; padding: 5px; margin: 5px 0;">  When Avro is selected as the Target storage format, the <code>TIMESTAMP</code> and <code>DATE</code> data types (which are not supported by Avro) are mapped to <code>VARCHAR(37)</code>. </div>	
INT1	TINYINT
INT2	SMALLINT
INT4	INT
INT8	BIGINT
NUMERIC	DECIMAL (p,s)
REAL4	FLOAT
REAL8	DOUBLE
STRING	VARCHAR (Length)
UINT1	SMALLINT

Qlik Replicatedata types	Hadoop data types
UINT2	INT
UINT4	BIGINT
UINT8	DECIMAL (20,0)
WSTRING	VARCHAR (Length)
BLOB	STRING
NCLOB	STRING
CLOB	STRING

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

Support for Partitions, Buckets and Skews To load data into tables with partitions, buckets or skews, you first need to perform the procedure described below.

To load data into tables with partitions, buckets or skews:

1. Create the tables in Hive with these attributes (partitions, buckets or skews) *prior* to running the task.
2. Add the following values to the

`hive.security.authorization.sqlstd.confwhitelist.append` property in the Hive configuration file:

If the target tables are partitioned:

```
|hive.exec.dynamic.partition|hive.exec.dynamic.partition.mode
```

If the target tables have buckets:

```
|hive.enforce.bucketing
```

If the target tables have skews:

```
|hive.mapred.supports.subdirectories
```



In some Hadoop Distributions, you may need to specify the value without the "hive" prefix. For example, |enforce.bucketing instead of |hive.enforce.bucketing.



If the value(s) already exist in the `hive.security.authorization.sqlstd.confwhitelist` property, you do not need to add them to the `hive.security.authorization.sqlstd.confwhitelist.append` property.

3. Set the **Target Table Preparation** task setting to **Truncate before loading** or **Do nothing**. For more information on these settings, see *Full Load Settings (page 1171)*.

To add a Hadoop target endpoint to Qlik Replicate:

1. In the Qlik Replicate console, click **Manage Endpoint Connections** to open the **Manage Endpoint Connections** dialog box.
For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints* (page 114).
2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the endpoint being used.
3. In the **Description** field, type a description that helps to identify the Hadoop endpoint. This is optional.
4. Select **Hadoop** as the endpoint **Type**.
5. In the **Security** section, do the following:
 1. To encrypt the data between the Replicate machine and HDFS, select **Use SSL**. In order to use SSL, first make sure that the [SSL](#) prerequisites described in *Prerequisites* (page 248) been met. In the **CA path** field, either specify the directory containing the CA certificate.
-OR-
Specify the full path to a specific CA certificate.
 2. Select one of the following authentication types:
 - **User name** - Select to connect to the Hadoop cluster with only a user name. Then, in the **User name** field, specify the name of a user authorized to access the Hadoop cluster.
 - *User name and password* - Select to connect to the Hadoop NameNode or to the Knox Gateway (when enabled - see below) with a user name and password. Then, in the *User name and Password* fields, specify the required user name and password. (page 783)
 - **Kerberos** - Select to authenticate against the Hadoop cluster using Kerberos. Replicate automatically detects whether Qlik Replicate Server is running on Linux or on Windows and displays the appropriate settings.



Note In order to use Kerberos authentication on Linux, the Kerberos client (workstation) package should be installed.

Qlik Replicate Server on Linux:

When Qlik Replicate Server is running on Linux, select either **Ticket** or [Keytab](#) from the **Kerberos options** drop-down list.

If you selected **Ticket**, select one of the following options:

- **Use global Kerberos ticket file** - Select this option if you want to use the same ticket for several Hadoop endpoints (source or target). In this case, you must make sure to select this option for each Hadoop endpoint instance that you define.
- **Use specific Kerberos ticket file** - Select this option if you want to use a different ticket file for each Hadoop endpoint (source or target). Then specify the ticket file name in the designated field.

This option is especially useful if you need to perform a task-level audit of Replicate activity (using a third-party tool) on the Hadoop NameNode. To set this up, define several instances of the same Hadoop endpoint and specify a unique Kerberos ticket file for each instance. Then, for each task, simply select a different Hadoop endpoint instance.



- You need to define a global Kerberos ticket file even if you select the **Use specific Kerberos ticket file** option. The global Kerberos ticket file is used for authentication when selecting a Hive endpoint, when testing the connection (using the **Test Connection** button), and when selecting which tables to replicate.
- When replicating from a Hadoop source endpoint to a Hadoop target endpoint, both endpoints must be configured to use the same ticket file.

For additional steps required to complete setup for Kerberos ticket-based authentication, see *Using Kerberos authentication* (page 792).

If you selected **Keytab**, provide the following information:

- **Realm:** The name of the realm in which your Hadoop cluster resides. For example, if the full principal name is `john.doe@EXAMPLE.COM`, then `EXAMPLE.COM` is the realm.
- **Principal:** The user name to use for authentication. The principal must be a member of the realm entered above. For example, if the full principal name is `john.doe@EXAMPLE.COM`, then `john.doe` is the principal.
- **Keytab file:** The full path of the Keytab file. The Keytab file should contain the key of the **Principal** specified above.

Qlik Replicate Server on Windows:


When Qlik Replicate Server is running on Windows, select one of the following:

- **Use the following KDC:** Select **Active Directory** (default) if your KDC is Microsoft Active Directory or select **MIT** if your KDC is MIT KDC running on Linux/UNIX.




When the Replicate KDC and the Hadoop KDC are in different domains, a relationship of trust must exist between the two domains.

- **Realm:** The name of the realm/domain in which your Hadoop cluster resides (where *realm* is the MIT term while *domain* is the Active Directory term).
- **Principal:** The user name to use for authentication. The principal must be a member of the realm/domain entered above.
- When **Active Directory** is selected - **Password:** The password for the principal entered above.
- When **MIT** is selected - **Keytab file:** The keytab file containing the principal entered above.


 When replicating from a Hadoop source endpoint to a Hadoop target endpoint, both endpoints must be configured to use the same parameters (KDC, realm, principal, and password).

If you are unsure about any of the above, consult your IT/security administrator. For additional steps required to complete setup for Kerberos authentication, see *Using Kerberos authentication* (page 792).


- **User name and password** - Select to connect to the Hadoop NameNode or to the Knox Gateway (when enabled - see below) with a user name and password. Then, in the **User name** and **Password** fields, specify the required user name and password.

 Consider the following:


- A user name and password is required to access the MapR Control System.
- This information is case sensitive.

 Make sure that the specified user has the required Hadoop access privileges. For information on how to provide the required privileges, see *Security requirements* (page 779).

6. If you need to access the Hortonworks Hadoop distribution through a Knox Gateway, select **Use Knox Gateway**. Then provide values for the following fields:


 To be able to select this option, first select **Use SSL** and then select **Password** from the **Authentication type** drop-down list.

- **Knox Gateway host** - The FQDN (Fully Qualified Domain Name) of the Knox Gateway host.
- **Knox port** - The port number to use to access the host. The default is "8443".
- **Knox Gateway path** - The context path for the gateway. The default is "gateway".


 The port and path values are set in the **gateway-site.xml** file. If you are unsure whether the default values have been changed, contact your IT department.


- **Cluster name** - The cluster name as configured in Knox. The default is "default".

7. In the **HDFS section**, select **WebHDFS**, **HttpFS** or **NFS** as the HDFS access method. If you are accessing MapR, it is recommended to use HttpFS.


 When the **Use Knox Gateway** option is selected, the **NameNode**, **HttpFS Host**, and **Port** fields described below are not relevant (and are therefore hidden).

- If you selected **WebHDFS**:
 - a. In the **NameNode** field, specify the IP address of the NameNode.


 *This is the Active node when High Availability is enabled (see below).*
 - b. Replicate supports replication to an HDFS High Availability cluster. In such a configuration, Replicate communicates with the Active node, but switches to the Standby node in the event of failover. To enable this feature, select the **High Availability** check box. Then, specify the FQDN (Fully Qualified Domain Name) of the Standby NameNode in the **Standby NameNode** field.
 - c. In the **Port** field, optionally change the default port (50070).
 - d. In the **Target Folder** field, specify where to create the data files on HDFS.
- If you selected **HttpFS**:
 - a. In the **HttpFS Host** field, specify the IP address of the HttpFS host.
 - b. In the **Port** field, optionally change the default port (14000).
 - c. In the **Target Folder** field, specify where to create the data files on HDFS.
- If you selected **NFS**:
 - a. In the **Target folder** field, enter the path to the folder located under the MapR cluster mount point. For example: `/mapr/my.cluster.com/data`
 - b. In order to do this, you first need to mount the MapR cluster using NFS. For information on how to do this, refer to the MapR help.

 *The **Target folder** name can only contain ASCII characters.*

8. In the **Hive Access section**, do the following:
 1. From the **Access Hive using** drop-down list, select one of the following options:

 *When the **Use Knox Gateway** option is selected, the **Host** and **Port** fields described below are not relevant (and are therefore hidden).*

- **ODBC** - Select this option to access Hive using an ODBC driver (the default). Then continue from the **Host** field.

 *If you select his option, make sure that the latest 64-bit ODBC driver for your Hadoop distribution is installed on the Qlik Replicate Server machine.*

- **HQL scripts** - When this option is selected, Replicate will generate HQL table creation scripts in the specified **Script folder**.





 *When this option is selected, the [target storage format](#) must be set to "Text".*

- **No Access** - When this option is selected, after the data files are created on HDFS, Replicate will take no further action.
2. In the **Host** field, specify the IP address of the Hive machine.
 3. In the **Port** field, optionally change the default port.
 4. In the **Database** field, specify the name of the Hive target database.



Setting advanced connection properties



The table below describes the settings in the **Advanced** tab.



Advanced tab settings


Setting	Description
File Format	Expand this section to specify or view the file format settings.
Target storage format	<p>Select one of the following target storage formats: Text (the default), Avro, ORC, Parquet, Sequence.</p> <div style="border: 1px solid #ccc; padding: 10px; margin-bottom: 10px;"> <p> For both regular tables and Replicate Control Tables, creating and storing the tables in text format (the default) allows data to be appended to them. This in turn reduces the number of files created on Hadoop, improves query performance, and reduces the number of Hive jobs running.</p> </div> <div style="border: 1px solid #ccc; padding: 10px; margin-bottom: 10px;"> <p> If Avro, ORC or Parquet is selected or if the target tables have skews/buckets, Replicate first converts the source data to a temporary sequence file and then runs a Hive process to convert the sequence file to the desired target format. As this will increase latency due to the additional Hive processes, it is recommended not use the aforementioned formats unless absolutely necessary.</p> </div> <div style="border: 1px solid #ccc; padding: 10px; margin-bottom: 10px;"> <p> When using the default text Serde (see below), new lines as part of the data values are not supported (due to an Apache Hadoop limitation). Although other Serdes may support new lines, best practice is to use Sequence as the target storage format.</p> </div> <div style="border: 1px solid #ccc; padding: 10px;"> <p> Unlike other binary formats that need to be converted to the desired target format (see above), when Sequence format is selected, the data is loaded directly to the target and stored in an external table (in sequence format).</p> <p>Note that Snappy compression is not available for sequence format.</p> </div> <p>See also: Prerequisites for using the Cloudera Distribution as a Hadoop target (page 777).</p>



10 Adding and managing target endpoints

Setting	Description
Control Tables storage format	<ul style="list-style-type: none"> • Text: This is the default method. • Same as the target storage format: This method is only recommended if, for whatever reason, you cannot use Text format. For example, this may be the case if you have an existing process that only knows how to read/interpret the Control Table data in a different format).
Use Default SerDe	Choose the SerDe interface to use when accessing the Hive database tables. The default is LazySimpleSerde .
Other SerDe	LazySimpleSerde creates the target files in delimited text file format. To create the target files in a different format, select the Other SerDe field and then specify the name of the SerDe that you want to use.
Field delimiter	<p>The delimiter that will be used to separate fields in the target file. The default is <code>\001</code>. Note that field delimiters should be ASCII characters unless they are non-printable control characters, in which case they should be in Octal format (e.g. <code>\247</code>).</p> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> <i>When using other SerDe:</i></p> <p>The default name for the field delimiter property is <code>field.delim</code>. If you selected Other SerDe and the specified SerDe uses a different property name (e.g. <code>separatorChar</code>), in addition to specifying the property value here, you also need to specify both the property name and its value in the SerDe properties field (e.g. <code>separatorChar=\t</code>).</p> </div>
Null value	<p>The value that will be used to indicate a null value in the target file.</p> <p>Example (where @ is the null value):</p> <p>mike,male,295678</p> <p>sara,female,@</p> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> <i>When using other SerDe:</i></p> <p>The default name for the null value property is <code>serialization.null.format</code>. If you selected Other SerDe and the specified SerDe uses a different property name (e.g. <code>nullChar</code>), in addition to specifying the property value here, you also need to specify both the property name and its value in the SerDe properties field (e.g. <code>nullChar=@</code>).</p> </div>

Setting	Description
Escape character	<p>When using LazySimpleSerDe: The escape character is used to escape the field delimiter character. When a field delimiter is escaped, it is interpreted as actual data, and not as a field delimiter.</p> <p>Example (where \ is the escape character and a comma is the field delimiter):</p> <p>sunroof\,power-steering</p> <p>When using Other SerDe: The escape character is used to escape the quote character.</p> <p>Example (where \ is the escape character and double quotes is the quote character):</p> <p>"\"sunroof, power-steering\""</p> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> <i>When using other SerDe:</i></p> <p>The default name for the escape character property is <code>escape.delim</code>. If you selected Other SerDe and the specified SerDe uses a different property name (e.g. <code>escapeChar</code>), in addition to specifying the property value here, you also need to specify both the property name and its value in the SerDe properties field (e.g. <code>escapeChar=</code>).</p> </div>
Record delimiter	<p>The <code>\n</code> delimiter is used to separate records (rows) in the target files. When using the default SerDe (LazySimpleSerde), the record delimiter cannot be changed.</p> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> <i>When using other SerDe:</i></p> <p>The default name for the record delimiter property is <code>line.delim</code>. If you selected Other SerDe and the specified SerDe uses a different property name (e.g. <code>recordChar</code>), in addition to specifying the property value here, you also need to specify both the property name and its value in the SerDe properties field (e.g. <code>recordChar=\n</code>).</p> </div>

Setting	Description
Quote character	<p>The quote character is used to escape the field delimiter character. When a field delimiter is escaped, it is interpreted as actual data, and not as a field delimiter. Note that the quote character is not available when using the default SerDe (LazySimpleSerde).</p> <p>Example (where double-quotes is the quote character):</p> <pre>"mike,male"</pre> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> <i>When using other SerDe:</i></p> <p><i>The default name for the quote character property is <code>quote.delim</code>. If you selected Other SerDe and the specified SerDe uses a different property name (e.g. <code>quoteChar</code>), in addition to specifying the property value here, you also need to specify both the property name and its value in the SerDe properties field (e.g. <code>quoteChar='</code>).</i></p> </div>
SerDe properties	<p>Enter the SerDe properties if Other SerDe is selected and the SerDe properties are not the same as the Hadoop defaults (<code>field.delim</code>, <code>serialization.null.format</code>, <code>escape.delim</code>, <code>line.delim</code>, <code>quote.delim</code>).</p> <p>The properties should be written using the following format:</p> <pre>"KEY1=VALUE1,KEY2=VALUE2,KEY3=VALUE3"</pre> <p>The list of properties should begin and end with a quotation mark.</p> <p>Example:</p> <pre>"separatorChar=\t,escapeChar={,quoteChar=' "</pre> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> <i>When " is specified as a value, it needs to be enclosed with quotation marks and escaped with a quotation mark, as follows: """"</i></p> </div>

Setting	Description
Add metadata header	<p>When the target storage format is set to Text, you can optionally add a header row to the data files. The header row can contain the source column names and/or the intermediate (i.e. Replicate) data types.</p> <p>Example of a target file with a header row when both With column names and With data types are selected:</p> <p>Position:DECIMAL(38,0),Color:VARCHAR(10)</p> <p>1,"BLUE"</p> <p>2,"BROWN"</p> <p>3,"RED"</p> <p>...</p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">  <i>This option is only available when "No Access" is selected as the Hive access method (in the General tab).</i> </div>
File Attributes	Expand this section to specify or view the file attributes.
Use Hadoop defaults	Select to work with the default block size of your Hadoop target.
Use this block size (MB)	Select to work with a different block size. The default value is 64.
Maximum file size	Specify the maximum file size of each target file. When the data reaches the maximum size, the file will be closed and written to the specified target folder.

Setting	Description
Compress files using	<p>Select the compression method to use on HDFS.</p> <div style="border: 1px solid #ccc; padding: 10px; margin-bottom: 10px;"> <p> <i>Cloudera ODBC drivers 2.5.20 or later do not support the Snappy compression method.</i></p> </div> <div style="border: 1px solid #ccc; padding: 10px;"> <p> <i>To use Snappy compression when the Setting advanced connection properties (page 785) is set to Avro, Parquet or Text, you must add the following values to the <code>hive.security.authorization.sqlstd.confwhitelist.append</code> property in the Hive configuration file:]</i></p> <ul style="list-style-type: none"> • For Avro: <code> hive.exec.compress.output avro.output.codec</code> • For Parquet: <code> hive.exec.compress.output parquet.compression</code> • For Text: <code> hive.exec.compress.output parquet.compression</code> <p><i>Note that in some Hadoop Distributions, compression will only work if you specify the value without the "hive" prefix. For example <code> exec.compress.output</code> instead of <code> hive.exec.compress.output</code>.</i></p> <p><i>If the value(s) already exist in the <code>hive.security.authorization.sqlstd.confwhitelist</code> property, you do not need to add them to the <code>hive.security.authorization.sqlstd.confwhitelist.append</code> property.</i></p> <p><i>See also: Prerequisites for using the Cloudera Distribution as a Hadoop target (page 777).</i></p> </div>
Change Processing	Expand this section to specify or view change processing settings.
Consider state idle when no changes have been processed for	Specify how long to wait before considering the state to be idle. In idle state, you can create files from data that has already been processed if the specified size and time conditions are met (see below).
File size reaches	Specify the minimum size of the data required to create a file in idle state.

Setting	Description
Elapsed time reaches	Specify the maximum time to wait before applying the changes in idle state.



*To facilitate rapid delivery of DDL messages, files are uploaded immediately, regardless of the specified **File size reaches** or **Elapsed time reaches** values.*

Preventing ODBC connection timeouts

The default query timeout value is 600 seconds, which should be sufficient for most situations. However, when loading very large tables, you may need to increase the value to prevent timeouts. This can be done using the following internal parameter:

`executeTimeout`

See below for instructions on setting internal parameters.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

Using Kerberos authentication

Whether Qlik Replicate Server is running on Linux or Windows, you can configure it to authenticate itself against the Hadoop cluster using Kerberos.

This requires you to perform the following steps on the Qlik Replicate machine before starting the Qlik Replicate Server.

Using Kerberos authentication on Linux

To use Kerberos authentication on Linux:



The commands described below should be issued under the "Attunity" user or under the user that was selected during the Replicate installation.

1. Obtain a valid TGT (Ticket-Granting Ticket) from the Kerberos KDC (Key Distribution Center) but save the TGT to a non-default cache file. Usually, a keytab file is used to perform non-interactive authentication to Kerberos.

Command Syntax:

```
kinit -kt [keytab_file] -c [cache_file_name] [principal_name]
```

2. This step is only required for the global Kerberos ticket file. Set the Kerberos cache environment variable (for Replicate to use later on).

To set the environment variable:

- a. Change the working directory to the Replicate "bin" directory by issuing the following command (assumes the default installation path):

```
cd /opt/attunity/replicate/bin
```

- b. Stop the Qlik Replicate Server services on the Linux by running:

```
/opt/attunity/replicate/bin/areplicate stop
```

3. Create a file named `site_arep_login.sh` in the Qlik Replicate bin folder.

- a. Add the following command to the file:

```
export KRB5CCNAME=cache_file_name
```

Example:

```
export KRB5CCNAME=/temp/kerberos/global.ticket
```

- b. Save the file and

- c. Start the Qlik Replicate Server services on the Linux by running:

```
/opt/attunity/replicate/bin/areplicate start
```

Now, whenever Qlik Replicate needs to use Kerberos authentication, it will perform the following operations:

- When **Use global Kerberos ticket file** is selected: Replicate will check whether the `KRB5CCNAME` environment variable is set and, if so, will use the ticket(s) inside the cache file specified by the environment variable.
- When **Use specific Kerberos ticket file** is selected:
 - During design-time (e.g. when selecting tables, testing the connection, etc.), Replicate will use the ticket(s) inside the cache file specified by the `KRB5CCNAME` environment variable.

10 Adding and managing target endpoints

- During runtime, Replicate will use the ticket file specified in the Hadoop endpoint settings.



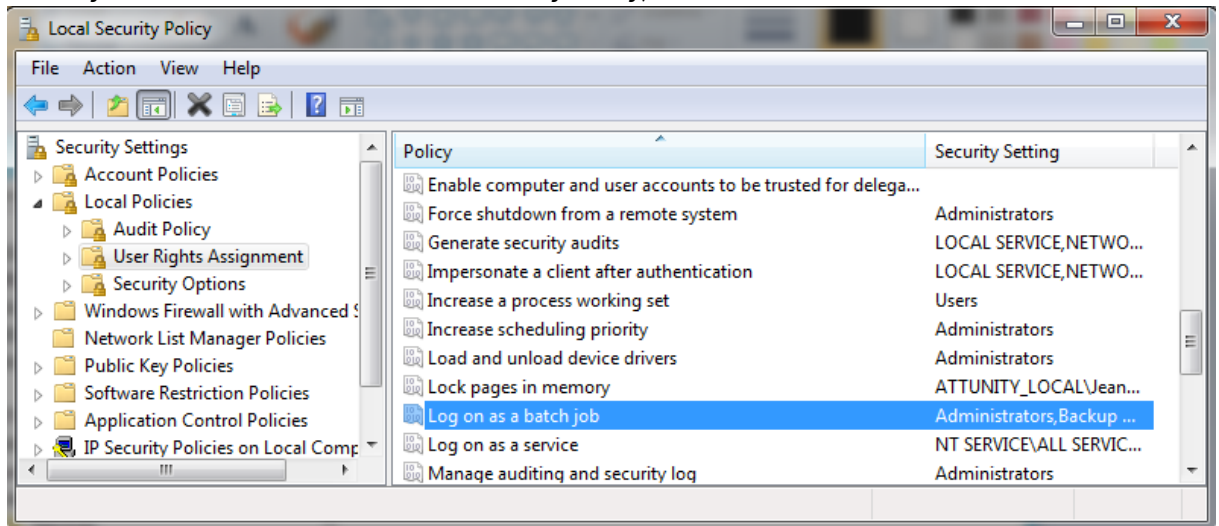
*If the ticket in the cache file expires or becomes invalid, repeating the `kinit` command shown in **Step 1** above will write a new TGT to the cache file and allow Qlik Replicate to continue working. This can be done without restarting the Qlik Replicate Server.*

Using Kerberos authentication on Windows

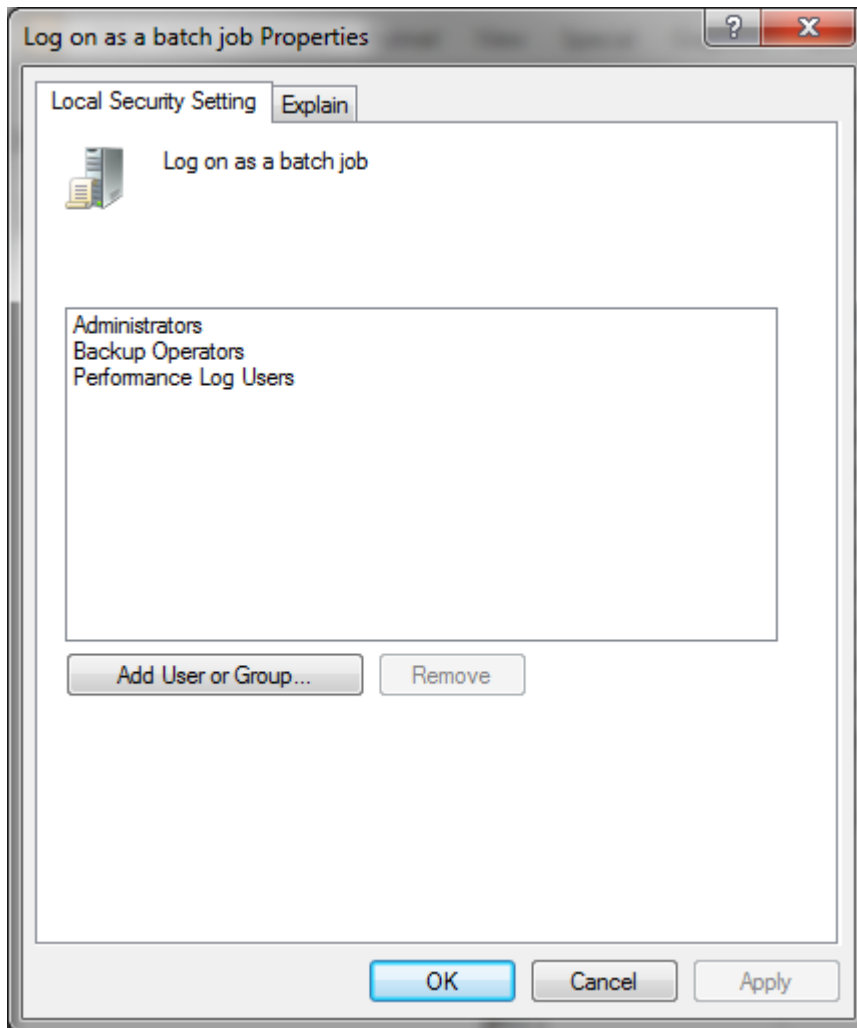
Before beginning, make sure that the impersonated user (principal) is granted read and write permissions on the Replicate **Data** directory (<product_dir>\Data by default) on the Qlik Replicate server. For Active Directory KDC, the impersonated user is the user configured in the user interface. For MIT KDC, this is the Windows user to which the MIT principal is mapped.

To set up Kerberos authentication on Windows, perform the following steps to ensure that the impersonated user (principal) has the **Log on as a batch job** privilege on the Qlik Replicate server.

1. On the Qlik Replicate Server machine, open the **Local Security Settings (Control Panel > System Security > Administrative Tools > Local Security Policy)**.



2. In the console tree, expand **Local Policies** and select **User Rights Assignments**.
3. In the details pane, double-click **Log on as a batch job**.
4. In the **Log on as a batch job Properties** dialog box, on the **Local Security Settings** tab, verify that the relevant user is listed. If the user is not listed, click **Add User or Group**, then add the user and click **OK**.



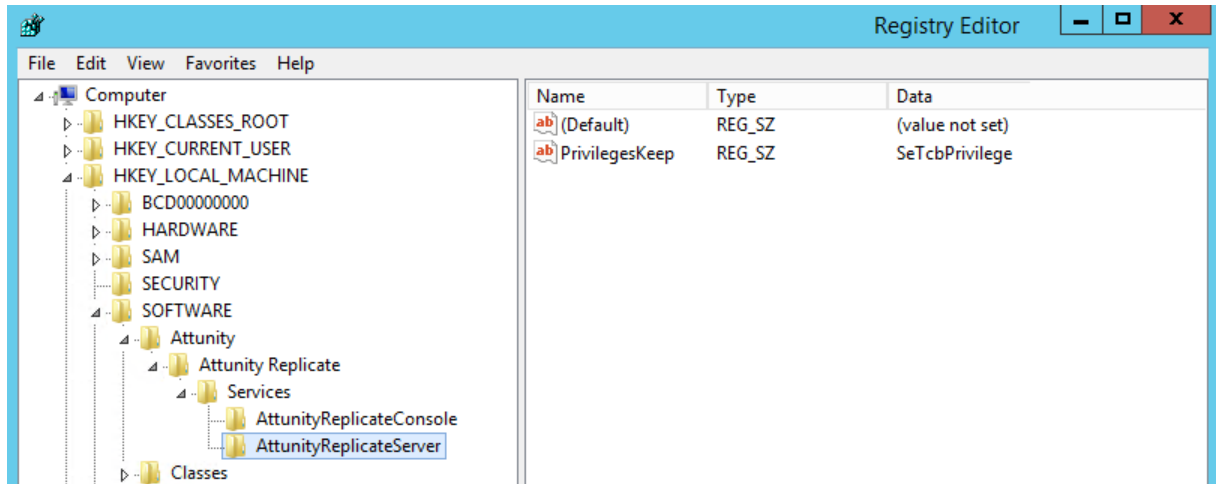
Your changes should take effect immediately.

MIT Kerberos

When MIT Kerberos is set in the endpoint, you need to perform the following steps to allow the Qlik Replicate server process to keep a specific privilege on startup. By default, Qlik Replicate server drops all privileges on startup. These steps are not required if you use Active Directory KDC.

1. Open the Windows registry (regedit.exe).
2. Browse to: HKEY_LOCAL_MACHINE\SOFTWARE\Qlik\Qlik Replicate\Services\AttunityReplicateServer

10 Adding and managing target endpoints



3. Modify the `PrivilegesKeep` string to include the value `SeTcbPrivilege`.
4. Close the **Registry Editor** window.
5. Restart the **Qlik Replicate Server** service.

10.21 Using Hortonworks Data Platform (HDP) as a target

This section describes how to set up and use Hortonworks Data Platform (HDP) as the target endpoint in a replication task.

In this section:

- *Prerequisites (page 797)*
- *Limitations (page 797)*
- *Change Data Partitioning (page 798)*
- *Security requirements (page 798)*
- *HDP Endpoint target data types (page 798)*
- *Setting general connection properties (page 799)*
- *Setting advanced connection properties (page 806)*
- *Using Kerberos authentication (page 809)*

Prerequisites

Before you begin to work with an HDP cluster as a target in Qlik Replicate, make sure that the following prerequisites have been met:

- **General:**
 - The HDFS or Amazon S3 storage must be accessible from the Qlik Replicate machine.
- **ODBC Access:**
 - The Hive Server must be accessible from the Qlik Replicate machine.
 - Cloudera ODBC driver for Apache Hive 2.6.9 or later must be installed on the Qlik Replicate machine.
- **SSL:** Before you can use SSL, you first need to perform the following tasks:
 - Configure each NameNode and each DataNode with an SSL certificate (issued by the same CA).
 - Place the CA certificate on the Replicate Server machine. The certificate should be a base64-encoded PEM (OpenSSL) file.
- **Permissions:** The user specified in the HDP target settings must have write permission for the specified HDFS or Amazon S3 target directory (depending on your [preferred storage type](#)).

Prerequisites for using a Linux ODBC driver

To use a Linux ODBC driver, make sure to:

- Install the latest 64-bit ODBC driver for HDP on the Replicate Server machine.
- After the driver is installed: Edit the <distribution>.hiveodbc.ini file as follows:
 - `DriverManagerEncoding=UTF-16`
 - `ODBCInstLib=libodbcinst.so`

See also *Setting general connection properties* (page 799).

Limitations

The following limitations apply:

- UPDATE/DELETE DMLs are not supported during change processing. If an UPDATE/DELETE DML was captured on the source, it will be ignored on the target and a warning will be written to the log. If the **Store Changes** option is enabled in the task settings, these records will be written to the Change Table.
- Limited LOB support only.
- Dropping columns and changing column data types or the data type length is not supported and will suspend the table in all cases *except* for data types that are mapped to STRING. Changes to the data type length of a data type mapped to STRING (e.g. VARCHAR(50) to VARCHAR(100)) will simply be ignored.
- The following Control Tables are not supported as they require UPDATE/DELETE operations, which are not supported:
 - **Replication Status** (requires UPDATE).
Name on target: `attrep_status`

- **Suspended Tables** (requires DELETE).

Name on target: `attrep_suspended_tables`

For more information on Control Tables, see *Control tables* (page 1169).

- Tables with partitions, buckets or skews are not supported.
- Table and column names can only contain ASCII characters.
- Column names cannot contain the '?' symbol. If needed, the '?' symbol can be replaced with a valid ASCII character [using a transformation](#).

Change Data Partitioning

When *Change Data Partitioning* (page 1177) is enabled, the Replicate Change Tables in Hive are partitioned by the `partition_name` column. Data files are uploaded to your [preferred storage provider](#), according to the [maximum size and time definition](#), and then stored in a directory under the Change Table directory. Whenever the specified partition timeframe ends, a partition is created in Hive, pointing to the target directory on your [preferred storage provider](#).

Information about the partitions is written to the [attrep_cdc_partitions](#) Control Table.

Prerequisites

The prerequisites for using Change Data Partitioning with the Hortonworks Data Platform (HDP) endpoint are as follows:

- The target file format must be set to Text or Sequence
- Hive access must be set to ODBC

Security requirements

- The HDP NameNode (and data nodes when using WebHDFS) must be accessible from the Qlik Replicate machine and the user specified in the HDP target settings must have write permission for the specified HDFS target directory.
- **Hive table permissions:** Replicate requires permissions to perform the following operations on Hive tables: CREATE, DROP, DESCRIBE, and ALTER table. ALTER table may also include RENAME table, ADD/RENAME column, and CREATE/DROP partitions on tables.

HDP Endpoint target data types

The following table shows the HDP endpoint target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).

Qlik Replicate and HDP data types

Qlik Replicatedata types	HDP data types
BOOL	BOOLEAN
BYTES	STRING
TIME	STRING
DATETIME	TIMESTAMP
DATE	DATE
INT1	TINYINT
INT2	SMALLINT
INT4	INT
INT8	BIGINT
NUMERIC	DECIMAL (p,s)
REAL4	FLOAT
REAL8	DOUBLE
STRING	VARCHAR (Length)
UINT1	SMALLINT
UINT2	INT
UINT4	BIGINT
UINT8	DECIMAL (20,0)
WSTRING	VARCHAR (Length)
BLOB	STRING
NCLOB	STRING
CLOB	STRING

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a Hortonworks Data Platform (HDP) target endpoint to Qlik Replicate:

1. In the Qlik Replicate console, click **Manage Endpoint Connections** to open the **Manage Endpoint Connections** dialog box.
For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints* (page 114).
2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the endpoint being used.

3. In the **Description** field, type a description that helps to identify the HDP endpoint. This is optional.
4. Select **Hortonworks Data Platform (HDP)** as the endpoint **Type**.
5. In the **Security** section, do the following:



These settings are relevant for HDFS storage and Hive only.

1. To encrypt the data between the Replicate machine and HDFS, select **Use SSL**. In order to use SSL, first make sure that the SSL prerequisites described in *Prerequisites (page 797)* been met. In the **CA path** field, specify one of the following:
 - The full path of a CA certificate file (in PEM format).
 - The directory containing the certificate files with hash names.
2. Select one of the following authentication types:
 - **User name** - Select to connect to the HDP cluster with only a user name. Then, in the **User name** field, specify the name of a user authorized to access the cluster.
 - **User name and password** - Select to connect to the HDP NameNode or to the Knox Gateway (when enabled) with a user name and password. Then, in the **User name** and **Password** fields, specify the required user name and password.
 - **Kerberos** - Select to authenticate against the HDP cluster using Kerberos. Replicate automatically detects whether Qlik Replicate Server is running on Linux or on Windows and displays the appropriate settings.



Note *In order to use Kerberos authentication on Linux, the Kerberos client (workstation) package should be installed.*

Qlik Replicate Server on Linux:

When Qlik Replicate Server is running on Linux, select either **Ticket** or [Keytab](#) from the **Kerberos options** drop-down list.

If you selected **Ticket**, select one of the following options:

- **Use global Kerberos ticket file** - Select this option if you want to use the same ticket for several HDP endpoints. In this case, you must make sure to select this option for each HDP endpoint instance that you define.
- **Use specific Kerberos ticket file** - Select this option if you want to use a different ticket file for each HDP endpoint. Then specify the ticket file name in the designated field.

This option is especially useful if you need to perform a task-level audit of Replicate activity (using a third-party tool) on the HDP NameNode. To set this up, define several instances of the same HDP endpoint and specify a unique Kerberos ticket file for each instance. Then, for each task, simply select a different HDP endpoint instance.



- You need to define a global Kerberos ticket file even if you select the **Use specific Kerberos ticket file** option. The global Kerberos ticket file is used for authentication when selecting a Hive endpoint, when testing the connection (using the **Test Connection** button), and when selecting which tables to replicate.
- When replicating from a Hadoop source endpoint to an HDP target endpoint, both endpoints must be configured to use the same ticket file.

For additional steps required to complete setup for Kerberos ticket-based authentication, see *Using Kerberos authentication* (page 809).

If you selected **Keytab**, provide the following information:

- **Realm:** The name of the realm in which your HDP cluster resides. For example, if the full principal name is `john.doe@EXAMPLE.COM`, then `EXAMPLE.COM` is the realm.
- **Principal:** The user name to use for authentication. The principal must be a member of the realm entered above. For example, if the full principal name is `john.doe@EXAMPLE.COM`, then `john.doe` is the principal.
- **Keytab file:** The full path of the Keytab file. The Keytab file should contain the key of the **Principal** specified above.

Qlik Replicate Server on Windows:

When Qlik Replicate Server is running on Windows, select one of the following:

- **Use the following KDC:** Select **Active Directory** (default) if your KDC is Microsoft Active Directory or select **MIT** if your KDC is MIT KDC running on Linux/UNIX.



When the Replicate KDC and the HDP KDC are in different domains, a relationship of trust must exist between the two domains.

- **Realm:** The name of the realm/domain in which your HDP cluster resides (where *realm* is the MIT term while *domain* is the Active Directory term).
- **Principal:** The user name to use for authentication. The principal must be a member of the realm/domain entered above.
- When **Active Directory** is selected - **Password:** The password for the principal entered above.
- When **MIT** is selected - **Keytab file:** The keytab file containing the principal entered above.




When replicating from a Hadoop source endpoint to an HDP target endpoint, both endpoints must be configured to use the same parameters (KDC, realm, principal, and password).

10 Adding and managing target endpoints


If you are unsure about any of the above, consult your IT/security administrator. For additional steps required to complete setup for Kerberos authentication, see *Using Kerberos authentication* (page 809).

- **User name and password** - Select to connect to the HDP NameNode or to the Knox Gateway (when enabled - see below) with a user name and password. Then, in the **User name** and **Password** fields, specify the required user name and password.


 *This information is case sensitive.*

 *Make sure that the specified user has the required HDP access privileges. For information on how to provide the required privileges, see *Security requirements* (page 798).*

6. If you need to access the HDP distribution through a Knox Gateway, select **Use Knox Gateway**. Then provide values for the following fields:

 *To be able to select this option, first select **Use SSL** and then select **Password** from the **Authentication type** drop-down list.*


- **Knox Gateway host** - The FQDN (Fully Qualified Domain Name) of the Knox Gateway host.
- **Knox port** - The port number to use to access the host. The default is "8443".
- **Knox Gateway path** - The context path for the gateway. The default is "gateway".

 *The port and path values are set in the **gateway-site.xml** file. If you are unsure whether the default values have been changed, contact your IT department.*


- **Cluster name** - The cluster name as configured in Knox. The default is "default".

7. In the **Storage** section, select the **HDFS** or **Amazon S3** storage type. Configure the **HDFS** or **Amazon S3** storage type settings depending on your selection as described in the following table.

HDFS storage

Option	Description
HDFS access method	Choose one of the following: <ul style="list-style-type: none">• WebHDFS• HttpFS <p> <i>When the Use Knox Gateway option is selected, the NameNode, HttpFS Host, and Port fields described below are not relevant (and are therefore hidden).</i></p>

10 Adding and managing target endpoints

Option	Description
When WebHDFS is the selected access method:	-
NameNode	Specify the IP address of the NameNode. <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">  <i>This is the Active node when High Availability is enabled (see below).</i> </div>
High Availability	Replicate supports replication to an HDFS High Availability cluster. In such a configuration, Replicate communicates with the Active node, but switches to the Standby node in the event of failover. To enable this feature, select the High Availability check box. Then, specify the FQDN (Fully Qualified Domain Name) of the Standby NameNode in the Standby NameNode field.
Port	Optionally, change the default port (50070).
Target Folder	Specify where to create the data files on HDFS.
When HttpFS is the selected access method:	-
HttpFS Host	Specify the IP address of the HttpFS host.
Port	Optionally, change the default port (14000).
Target Folder	Specify where to create the data files on HDFS.

Amazon S3 storage


Option	Description
Bucket name	The name of your Amazon S3 bucket.
Bucket region	The region where your bucket is located. It is recommended to leave the default (Auto-Detect) as it usually eliminates the need to select a specific region. However, due to security considerations, for some regions (for example, AWS GovCloud) you might need to explicitly select the region. If the region you require does not appear in the regions list, select Other and set the code using the regioncode internal parameter in the endpoint's Advanced tab. For a list of region codes, see the Region availability section in: https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Concepts.RegionsAndAvailabilityZones.html

Option	Description
Access options	<p>Choose one of the following:</p> <ul style="list-style-type: none"> • Key pair Choose this method to authenticate with your Access Key and Secret Key. • IAM Roles for EC2. Choose this method if the machine on which Qlik Replicate is installed is configured to authenticate itself using an IAM role. For more information about this access option, see: http://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles.html • Security Token Service (STS) Choose this method to authenticate using SAML 2.0 with Active Directory Federation Services. For more information about this access option, see: https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles_providers_saml.html
When Key pair is the access option:	-
Access key	Enter the access key information for Amazon S3.
Secret key	Enter the secret key information for Amazon S3.
When Security Token Service (STS) is the access option:	-
ADFS URL	The URL to an Active Directory Federation Services page, responsible for returning a SAML claims document to be sent over to AWS.
AD principal name	The principal (user) name to use when identifying against ADFS The format should be: <code>user.name@domain</code>


Option	Description
AD principal password	The principal password to use when identifying against ADFS
IdP ARN	The Amazon Resource Name (ARN) of the Active Directory issuing the SAML claims document. This is required as it enables AWS to identify the signer of the SAML document and verify its signature.
SAML Role ARN	The Amazon Resource Name (ARN) of the specific role the returned credentials should be assigned.
Switch role after assuming SAML role	Use this option to switch role after authentication. For more information, see: https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles_common-scenarios_aws-accounts.html When this option is selected, the following information is required: Role ARN: The ARN associated with the target role. Role external ID: The value of the external ID condition in the target role's trust policy.
For all access options:	-
Target folder	Enter the target folder in your Amazon S3 bucket.

8. In the **Hive Access section**, do the following:

- From the **Access Hive using** drop-down list, select one of the following options:

 When the **Use Knox Gateway** option is selected, the **Host** and **Port** fields described below are not relevant (and are therefore hidden).

- **ODBC** - Select this option to access Hive using an ODBC driver (the default). Then continue with the [Host field](#).

 If you select his option, make sure that the latest 64-bit ODBC driver for your Hadoop distribution is installed on the Qlik Replicate Server machine.

- **HQL scripts** - When this option is selected, Replicate will generate HQL table creation scripts in the specified **Script folder**.


 When this option is selected, the [target storage format](#) must be set to "Text".

- **No Access** - When this option is selected, after the data files are created on HDFS, Replicate will take no further action.


2. In the **Host** field, specify the IP address of the Hive machine.
3. In the **Port** field, optionally change the default port.
4. In the **Database** field, specify the name of the Hive target database.

Setting advanced connection properties

The table below describes the settings in the **Advanced** tab.

Advanced tab settings	
Setting	Description
File Format	Expand this section to specify or view the file format settings.
Target storage format	<p>Select the target storage format: Text (the default) or Sequence.</p> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p> <i>When using Text as the target storage format, new lines as part of the data values are not supported (due to an Apache Hadoop limitation). Best practice is to use Sequence as the target storage format.</i></p> </div>
Field delimiter	The delimiter that will be used to separate fields in the target file. The default is <code>\001</code> . Note that field delimiters should be ASCII characters unless they are non-printable control characters, in which case they should be in Octal format (e.g. <code>\247</code>).
Null value	<p>The value that will be used to indicate a null value in the target file.</p> <p>Example (where @ is the null value):</p> <pre>mike,male,295678 sara,female,@</pre>
Escape character	<p>When using LazySimpleSerde: The escape character is used to escape the field delimiter character. When a field delimiter is escaped, it is interpreted as actual data, and not as a field delimiter.</p> <p>Example (where \ is the escape character and a comma is the field delimiter):</p> <pre>sunroof\,power-steering</pre> <p>Example (where \ is the escape character and double quotes is the quote character):</p> <pre>"\"sunroof, power-steering\""</pre>

10 Adding and managing target endpoints

Setting	Description
Add metadata header	<p>When the target storage format is set to Text, you can optionally add a header row to the data files. The header row can contain the source column names and/or the intermediate (i.e. Replicate) data types.</p> <p>Example of a target file with a header row when both With column names and With data types are selected:</p> <p>Position:DECIMAL(38,0),Color:VARCHAR(10)</p> <p>1,"BLUE"</p> <p>2,"BROWN"</p> <p>3,"RED"</p> <p>...</p>
File Attributes	Expand this section to specify or view the file attributes.
Maximum file size	Specify the maximum file size of each target file. When the data reaches the maximum size, the file will be closed and written to the specified target folder.
Compress files using	Select the compression method to use on HDFS.
Change Processing	Expand this section to specify or view change processing settings.
Apply/store changes when:	Specify how long to wait before applying or storing the changes.
File size reaches	Specify the minimum size of the data file required to apply or store the changes.
Elapsed time reaches	Specify the maximum time to wait before applying or storing the changes.
<div style="border: 1px solid #ccc; padding: 10px; margin-bottom: 10px;">  <p><i>To facilitate rapid delivery of DDL messages, files are uploaded immediately, regardless of the specified File size reaches or Elapsed time reaches values.</i></p> </div>	
Proxy Server	Expand this section to specify the settings required when the storage (Amazon S3/HDFS) is accessed via a proxy server.
Use proxy server	Select this option to access the storage via a proxy server.
Host name	The host name of the proxy server.
Port	The port via which to access the proxy server.

10 Adding and managing target endpoints

Setting	Description
User name	The user name for accessing the proxy server.
Password	The password for accessing the proxy server.
Scheme	Select which protocol to use to access the server (HTTP or HTTPS). In order to use HTTPS when the Storage type field is set to Amazon S3 , you must first install the CA certificate that signed the proxy's certificate on the Replicate Server machine, as follows: <ul style="list-style-type: none">• On Windows: Add the CA certificate to the Trusted Root Certification Authorities store of Local Computer• On Linux: Add the CA certificate to /etc/pki/tls/certs/ca-bundle.crt
SSL CA Path	When HDFS is the Storage type and HTTPS is the selected Scheme , specify the location of the CA file on the Replicate Server machine .

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

Using Kerberos authentication

Whether Qlik Replicate Server is running on Linux or Windows, you can configure it to authenticate itself against the HDP cluster using Kerberos.

This requires you to perform the following additional steps on the Qlik Replicate machine.

Using Kerberos authentication on Linux

To use Kerberos authentication on Linux:



The commands described below should be issued under the "Attunity" user or under the user that was selected during the Replicate installation.

1. Obtain a valid TGT (Ticket-Granting Ticket) from the Kerberos KDC (Key Distribution Center) but save the TGT to a non-default cache file. Usually, a keytab file is used to perform non-interactive authentication to Kerberos.

Command Syntax:

```
kinit -kt [keytab_file] -c [cache_file_name] [principal_name]
```

2. This step is only required for the global Kerberos ticket file. Set the Kerberos cache environment variable (for Replicate to use later on).

To set the environment variable:

- a. Change the working directory to the Replicate "bin" directory by issuing the following command (assumes the default installation path):

```
cd /opt/attunity/replicate/bin
```

- b. Stop the Qlik Replicate Server services on the Linux by running:

```
/opt/attunity/replicate/bin/areplicate stop
```

3. Create a file named `site_arep_login.sh` in the Qlik Replicate bin folder.

- a. Add the following command to the file:

```
export KRB5CCNAME=cache_file_name
```

Example:

```
export KRB5CCNAME=/temp/kerberos/global.ticket
```

- b. Save the file and
- c. Start the Qlik Replicate Server services on the Linux by running:

```
/opt/attunity/replicate/bin/areplicate start
```

Now, whenever Qlik Replicate needs to use Kerberos authentication, it will perform the following operations:

- When **Use global Kerberos ticket file** is selected: Replicate will check whether the `KRB5CCNAME` environment variable is set and, if so, will use the ticket(s) inside the cache file specified by the environment variable.
- When **Use specific Kerberos ticket file** is selected:
 - During design-time (e.g. when selecting tables, testing the connection, etc.), Replicate will use the ticket(s) inside the cache file specified by the `KRB5CCNAME` environment variable.
 - During runtime, Replicate will use the ticket file specified in the HDP endpoint settings.



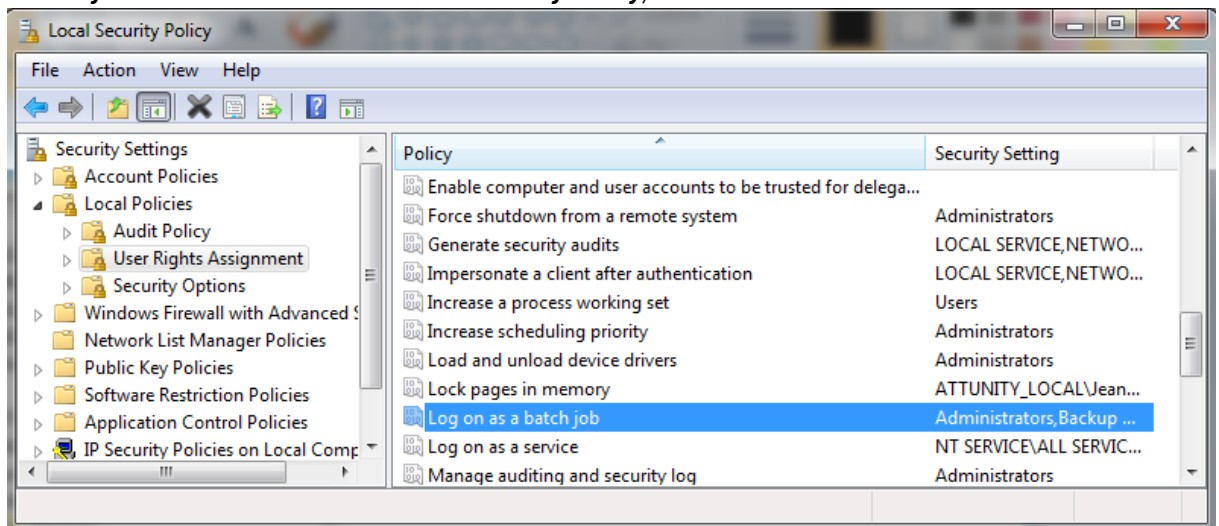
If the ticket in the cache file expires or becomes invalid, repeating the `kininit` command shown in **Step 1** above will write a new TGT to the cache file and allow Qlik Replicate to continue working. This can be done without restarting the Qlik Replicate Server.

Using Kerberos authentication on Windows

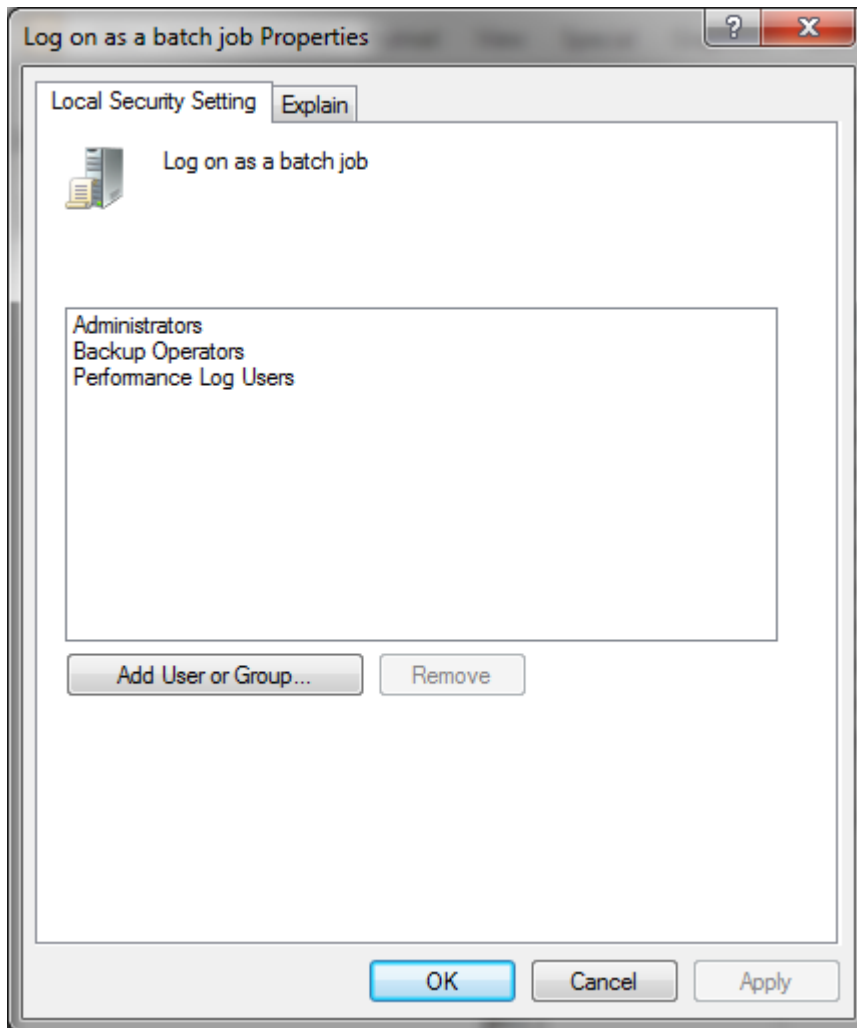
Before beginning, make sure that the impersonated user (principal) is granted read and write permissions on the Replicate **Data** directory (<product_dir>\Data by default) on the Qlik Replicate server. For Active Directory KDC, the impersonated user is the user configured in the user interface. For MIT KDC, this is the Windows user to which the MIT principal is mapped.

To set up Kerberos authentication on Windows, perform the following steps to ensure that the impersonated user (principal) has the **Log on as a batch job** privilege on the Qlik Replicate server.

1. On the Qlik Replicate Server machine, open the **Local Security Settings (Control Panel > System Security > Administrative Tools > Local Security Policy)**.



2. In the console tree, expand **Local Policies** and select **User Rights Assignments**.
3. In the details pane, double-click **Log on as a batch job**.
4. In the **Log on as a batch job Properties** dialog box, on the **Local Security Settings** tab, verify that the relevant user is listed. If the user is not listed, click **Add User or Group**, then add the user and click **OK**.



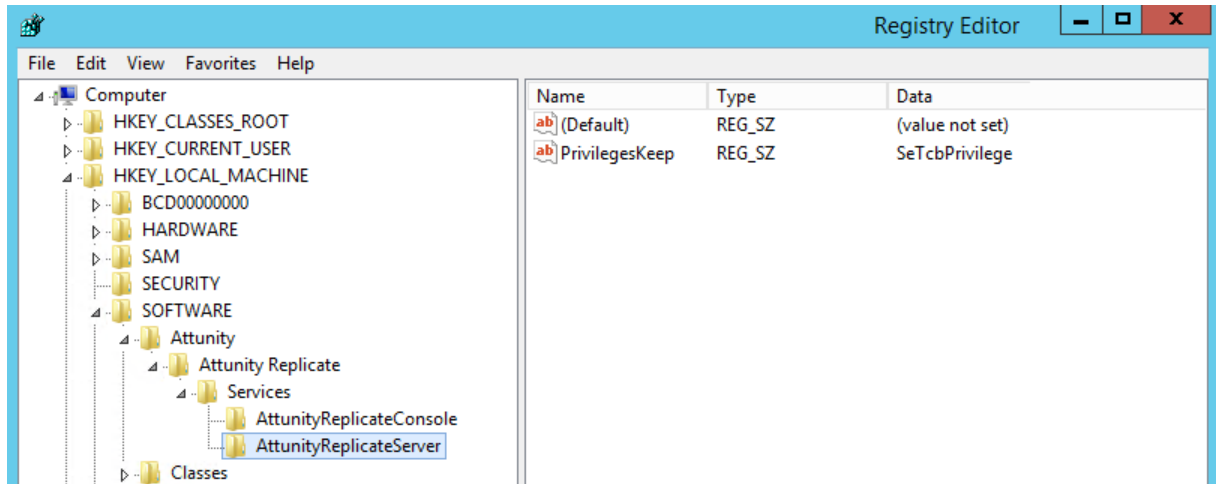
Your changes should take effect immediately.

MIT Kerberos

When MIT Kerberos is set in the endpoint, you need to perform the following steps to allow the Qlik Replicate server process to keep a specific privilege on startup. By default, Qlik Replicate server drops all privileges on startup. These steps are not required if you use Active Directory KDC.

1. Open the Windows registry (regedit.exe).
2. Browse to: HKEY_LOCAL_MACHINE\SOFTWARE\Qlik\Qlik Replicate\Services\AttunityReplicateServer

10 Adding and managing target endpoints



3. Modify the `PrivilegesKeep` string to include the value `SeTcbPrivilege`.
4. Close the **Registry Editor** window.
5. Restart the **Qlik Replicate Server** service.

10.22 Using IBM DB2 for z/OS as a target

This topic describes how to use IBM DB2 for z/OS connectivity to connect to a target endpoint.

In this section:

- *Prerequisites (page 814)*
- *Limitations and considerations (page 816)*
- *Permissions (page 817)*
- *Supported target data types (page 817)*
- *Setting general connection properties (page 819)*
- *Setting advanced connection properties (page 820)*

Prerequisites

The following section describes the prerequisites for working with Qlik Replicate and the IBM DB2 for z/OS target endpoint.


Both the IBM Data Server Driver for ODBC and CLI driver and the IBM Data Server Driver for JDBC and SQLJ driver need to be installed on the Replicate machine. When installing on Linux, you can choose between installing the full ODBC client package or installing just the data server client.

IBM Data Server Driver for ODBC and CLI driver setup

Supported drivers

Replicate supports driver versions 11.5.6 and 11.5.8 only. The setup instructions are for version 11.5.8.

Replicate on Windows


1. Download the IBM Data Server Client 11.5.8 (**v11.5.8_ntx64_client.exe**) from  [IBM Fix Central](#).
2. Install it on the Qlik Replicate Server machine.

Replicate on Linux

When installing on Linux, you can choose between installing the full ODBC client package or installing just the data server client.

Linux ODBC client package

If you prefer, you can install the data server client only as described in *Linux data server client (page 815)* below.

1. Download the IBM Data Server Driver for ODBC and CLI 11.5.8 (**v11.5.8_linuxx64_server_dec.tar.gz**) from  [IBM Fix Central](#).
2. Stop the Replicate service:

```
./areplicate stop
```
3. Optionally, confirm that the service has stopped:

```
./areplicate status
```
4. Install **v11.5.8_linuxx64_server_dec.tar.gz** and choose the “CLIENT” option.
5. If the Replicate machine does not have a DB2 instance, create a DB2 instance by running the following commands:

```
adduser <db2_instance_name>  
/opt/ibm/db2/v11.5/instance/db2icrt <db2_instance_name>
```
6. Change the working directory to `<Replicate_Install_Dir>/bin`.
7. Add the DB2 driver location to the Linux library path.
To do this, add the following line to the **site_arep_login.sh** file as follows:

```
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/path/lib
```


where path is the CLIENT path to the driver.
Example (with db2clnt1 as the instance name):

```
export LD_LIBRARY_PATH=$LD_LIBRARY_  
PATH:/home/db2clnt1/sql1lib/lib:/home/db2clnt1/sql1lib/lib64/gskit  
export DB2INSTANCE=db2clnt1
```
8. Optionally, confirm that the driver location was copied:

```
cat site_arep_login.sh
```

9. Add the following entry to the `odbcinst.ini` file under `/etc`:
[IBM DB2 ODBC DRIVER]
Driver = /home/db2c1nt1/sqllib/lib/libdb2o.so
fileusage=1
dontd1close=1
10. Start the Replicate service:
`./areplicate start`
11. Optionally, confirm that the service has started:
`./areplicate status`

Linux data server client

1. Download **v11.5.8_linuxx64_client.tar.gz** from  [IBM Fix Central](#) and extract the files.
2. Stop the Replicate service:
`./areplicate stop`
3. Optionally, confirm that the service has stopped:
`./areplicate status`
4. Change the working directory to `/opt/client` and run `db2_install`.
5. Create an RSP file (see **Sample RSP file** below) and then run the following command (from `/opt/client`):
`./db2setup -r /file-path/client.rsp`
6. Add the following entry to the **odbcinst.ini** file under `/etc`:
[IBM DB2 ODBC DRIVER]
Driver = /opt/ibm/db2/v11.5/lib64/libdb2o.so
fileusage=1
dontd1close=1
7. Start the Replicate service:
`./areplicate start`
8. Optionally, confirm that the service has started:
`./areplicate status`

Sample RSP file

```
*-----  
* Generated response file used by the DB2 Setup wizard  
* generation time: 2/18/22 4:10 PM  
*-----  
* Product Installation  
LIC_AGREEMENT      = ACCEPT  
PROD               = CLIENT  
FILE               = /opt/ibm/db2/V11.5  
INSTALL_TYPE      = TYPICAL  
*-----  
* Instance properties  
*-----  
INSTANCE          = inst1  
inst1.TYPE        = client  
* Instance-owning user  
inst1.NAME        = attunity  
inst1.UID         = 1000  
inst1.GROUP_NAME  = attunity
```

```
inst1.HOME_DIRECTORY      = /home/attunity
inst1.AUTOSTART           = NO
*-----
*  Installed Languages
*-----
LANG                       = EN
```

IBM Data Server Driver for JDBC and SQLJ driver setup

You need to install JDBC driver 4.31.10 on the Replicate machine. The driver is part of Db2 Version 11.5 Mod 6 Fix Pack 0.

Supported drivers

Replicate supports driver version 11.5.6 only.

Replicate on Windows

1. Download **v11.5.6_jdbc_sqlj.tar.gz** from [Fix Central on the IBM web site](#).
2. Extract the contents and copy the **db2jcc4.jar** file to <REPLICATE-INSTALL-DIR>\java
The default location is C:\Program Files\Attunity\Replicate\java.

Replicate on Linux

1. Download **v11.5.6_jdbc_sqlj.tar.gz** from [Fix Central on the IBM web site](#).
2. Extract the contents and copy the **db2jcc4.jar** file to <REPLICATE-INSTALL-DIR>/java.
The default location is /opt/attunity/replicate/java.

Application compatibility

The APPLCOMPAT bind option must be set to **V12R1**.

Limitations and considerations

When defining a task with an IBM DB2 for z/OS target endpoint, the following limitations apply:

- When replicating from an IBM DB2 for z/OS source, data type homogeneity will not be preserved.
- The **Explicit schema** option in the endpoint settings' **Advanced** tab is not supported in the following cases:
 - Source endpoints that do not have schemas (for example, Salesforce and File).
 - The **Allow unlimited LOB size** task setting is enabled.
 - The **Limit LOB size to** value in the task settings' **Target Metadata** tab is 16 KB or greater, and the **Change Processing Mode** in the task settings' **Change Processing Tuning** tab is set to **Transactional apply**.
- The replicated table row size must not exceed the size limit of the target table space. This also applies to internal tables created by Replicate, such as the Net Changes table, which is required for Batched Optimized Apply mode.



When both the **Load data using ZLOAD** option in the endpoint settings' **Advanced** tab and the **Limit LOB size to** option in the task settings' **Target Metadata** tab are enabled, the allowed row size includes LOB columns. When the **Load data using ZLOAD** option is disabled, the row size does not include LOB columns.

- When replicating tables with BLOB columns in Batch optimized apply mode, the **Limit LOB size to** value in the task settings' **Target Metadata** tab must not exceed 16 KB. If it does, the updated BLOB will be truncated to 16 KB when capturing changes.
- To replicate tables with column names that contain non-English letters, set the following internal parameter to `false: ignoreTargetExtraColumns`



Replication of column names with non-English letters is not supported for tables with LOB columns.

- Null values that are replicated into non-nullable columns are not handled properly when the **Load data using ZLOAD** option in the endpoint settings' **Advanced** tab is enabled. In such cases, you can define a transformation on non-nullable columns, as in the following example:

```
ifnull($c, "")
```

This transformation sets an empty value in non-nullable column 'c' when the data is null.

Permissions

The user defined in the endpoint settings must have the following permissions on the target database:

- Table permissions:
 - LOAD table (ZLOAD)
 - SELECT table
 - CREATE table
 - LIST tables
 - CREATE alias (only required for replicating tables names with non-English letters)
- Schema permissions:
 - LIST schemas
 - SELECT on SYSIBM.SYSDUMMY1 - Only required when configuring the **Current user schema** option in the endpoint settings' **Advanced** tab.

Supported target data types

The following table shows the IBM DB2 for z/OS target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).

Supported data types

Qlik Replicate Data Types	IBM DB2 z/OS Data Types
STRING	If length ≤ 32704: VARBINARY (Length) If length > 32704: BLOB (Length)
WSTRING	If length ≤ 16352: VARGRAPHIC (Length) If length > 16352: DBCLOB (Length)
UINT1	SMALLINT
UINT2	INTEGER
UINT4	BIGINT
UINT8	BIGINT
INT1	SMALLINT
INT2	SMALLINT
INT4	INTEGER
INT8	BIGINT
TIMESTAMP	If scale < 7: TIMESTAMP If scale 7-12: VARCHAR (37)
TIME	TIME
DATE	DATE
REAL4	REAL
REAL8	DOUBLE

Qlik Replicate Data Types	IBM DB2 z/OS Data Types
NUMERIC	If precision ≤ 31 and scale ≤ 31 : DECIMAL (p,s) If precision >31 and scale >31 : VARCHAR (45)
BOOLEAN	CHAR (5)
BYTES	If length ≤ 32704 : VARBINARY (Length) If length > 32704 : BLOB (Length)
CLOB	CLOB (2G)
BLOB	BLOB (2G)
NCLOB	DBCLOB (1G)

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add an IBM DB2 for z/OS target endpoint to Qlik Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog. Then click the **New Endpoint Connection** button.
2. In the **Name** field, specify a display name for your IBM DB2 for z/OS endpoint. This can be any name that will help to identify the endpoint being used.
3. Optionally, in the **Description** field, enter a description.
4. Select **Target** as the endpoint **Role**.
5. Select **IBM DB2 for z/OS** as the endpoint **Type**.
6. Configure the following properties:

- **Server (system):** The host name or IP address of the DB2 database server.



*If the **Server** is a parallel SysPlex and data sharing members reside on multiple LPARs, the host address may specify a DVIPA - a Dynamic Virtual IP Address (to utilize the system redundancy and load-balancing in Replicate processing). In this case, the port numbers must be identical for all members.*

- **Port:** The port through which to access the database. The default is 50,000.
 - **User Name:** The **User name** of an authorized user of the specified IBM DB2 for z/OS database. For a list of the permissions that need to be granted to this user, see *Permissions (page 817)*.
 - **Password:** The **Password** for the specified user.
 - **Location:** The **Location** should be the DB2 location name defined during the installation. This should be a relational database management system under z/OS, which can be either a subsystem or a group connection. This is the logical name which serves applications in order to designate resources managed by this system, either using the SQL CONNECT instruction, or placing it as a qualifier of a table (preceding the schema name). To see the location name, use the “-DIS DDF” DB2 command (option 7 under the DB2I panel in ISPF), or look in message DSNL004I in the job log of the <ssid>MSTR address space.
7. Click **Test Connection** to make sure that Replicate is able to establish a connection to the specified location and then (if the connection is successful), either click **Save** or modify the settings in the **Advanced** tab.

Setting advanced connection properties

In the **Advanced** tab, you can set the following properties:

- **Load data using ZLOAD:** Select to load data into the Replicate Net Changes table using the ZLOAD command. Replicate will then load the data from the Net Changes table into the target tables. This should improve performance when working in the Batch optimized apply change processing mode.
 - **Max file size (MB):** The maximum size (in MB) of each file to load using ZLOAD. The default value is 32 MB. If you encounter performance issues, try increasing or decreasing this value. The optimal value will depend on the number of files being uploaded in parallel as well as your specific environment.
- **Additional ODBC connection properties:** Specify any additional ODBC connection parameters that you want to use.
- **ODBC driver:** Should be **IBM DB2 ODBC Driver**.
- **Alternate schema:** If no source schema exists and no target schema is defined in the task settings, the alternate schema will be used for the target data. It will be used for the Replicate Control tables if no target table schema or Control table schema are defined in the task settings.
Select either:
 - **Current user schema** to create (and/or update) the tables in the current user schema.
-OR-
 - **Explicit schema** to create (and/or update) the tables in a specific schema. Then specify the **Schema name**.



This option is not supported with source endpoints that do not have schemas (for example, Salesforce and File).

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.23 Using Kafka as a target

This section describes how to set up and use Kafka as a target endpoint in a replication task. In a task with a Kafka target endpoint, each source record is transformed into a message which is then written (with an optional message key) to a partition in the specified topic.



Although the Kafka target endpoint supports working with Confluent Cloud, customers who are not already using it this way should use the dedicated Confluent Cloud target endpoint instead.

In this section:

- *Transaction processing by the consumer (page 823)*
- *Prerequisites (page 824)*
- *Limitations and considerations (page 824)*
- *Kafka target data types (page 825)*
- *Setting general connection properties (page 827)*
- *Setting advanced connection properties (page 835)*
- *Working with Confluent Cloud (page 836)*
- *The Replicate envelope (page 836)*
- *Metadata and data messages (page 838)*
- *Using Kerberos authentication on Windows (page 842)*

Transaction processing by the consumer

When configuring the Qlik Replicate Kafka endpoint, users can configure various settings that affect where messages are published within the Kafka infrastructures (topics/partitions).

During a task's CDC stage, committed changes that are detected by the Qlik Replicate source endpoint are grouped by transaction, sorted internally in chronological order, and then propagated to the target endpoint. The target endpoint can handle the changes in various ways such as applying them to the target tables or storing them in dedicated Change Tables.

Each CDC message has both a transaction ID as well as change sequence. As the change sequence is a monotonically growing number, sorting events by change sequence always achieves chronological order. Grouping the sorted events by transaction ID then results in transactions containing chronologically sorted changes.

However, as Kafka is a messaging infrastructure, applying changes is not feasible while storing changes in tables is meaningless. The Replicate Kafka endpoint, therefore, takes a different approach, which is to report all transactional events as messages.

How it works

Each change in the source system is translated to a data message containing the details of the change including the transaction ID and change sequence in the source. The data message also includes the changed columns before and after the change. As explained above, the order in which the Kafka target writes the messages is the same as order of changes within each transaction.

Once a data message is ready to be sent to Kafka, the topic and partition it should go to are determined by analyzing the [endpoint settings](#) as well as potentially [transformation settings](#). For example, the user might decide to configure the endpoint in such a way that every table is sent to a different topic and set the partition strategy to "Random", meaning that each message (within the same table) will be sent to a different partition.

Transaction consistency from a consumer perspective

If maintaining transaction consistency is important for the consumer implementation, it means that although the transaction ID exists in all data messages, the challenge is to gather the messages in a way that would facilitate identifying a whole transaction. An additional challenge is getting the transaction in the original order they were committed, which could be an even greater challenge if transactions are spread across multiple topics and partitions.

The simplest way of achieving the above goal is to direct Replicate to a specific topic and a specific partition (in the [endpoint settings](#)). This means that all data messages will end up in a single partition, thus guaranteeing ordered delivery both of transactions *and* of changes within a transaction. The consuming application could then consume messages - accumulating a transaction in some intermediate memory buffer - and when a new transaction ID is detected, mark the previous transaction as completed.

Although the simple way may work, it's not very efficient at the task level as all messages end up in the same topic and partition, not necessarily utilizing the full parallelism of the Kafka cluster. This may be a non-issue if there are multiple tasks, each taking advantage of a different topic/partition. In such as scenario, the gathering of messages from those tasks may very well utilize the cluster optimally.

The more generic way where data may be spread over multiple topics and partitions means that some intermediate buffer such as memory, a table in a relational database, or even other Kafka topics would need to be used to collect information about transactions. Then, the transactions would need to be rebuilt by periodically (every few minutes/hours) sorting the events collected from Replicate's Kafka output by the change sequence and grouping them by transaction ID.

Prerequisites

Before you can use Kafka as a target endpoint in a Replicate task, the following prerequisites must be met:

- Open TCP ports to all the brokers from the Replicate Server machine
- Set permissions that will allow Qlik Replicate to write to the target topics. One way to do this is to use the Kafka ACLs script (`kafka-acls`).
- Either create a topic named **attrep_apply_exceptions** before starting the replication task or configure the brokers with `auto.create.topics.enable=true`.
Note that if this topic does not exist, the task will always fail when it encounters a data error, regardless of the [error handling policy](#).
For a description of the **attrep_apply_exceptions** table, see *Apply exceptions (page 1298)*

Limitations and considerations

When defining a task with Kafka as the target endpoint, the following limitations apply:

- The Kafka target endpoint does not support unlimited LOB size. Therefore, when replicating from source tables with LOB columns, *do not* select the **Allow unlimited LOB size** option.
For more information on defining LOB settings, see *Target metadata (page 1166)*.
- **Batch optimized apply** mode is not supported. If this mode is set, the task will automatically switch to **Transactional apply** mode and issue an appropriate warning.
For more information on these modes, see *Change Processing Tuning (page 1181)*.
- Store Changes mode is not supported.
For more information on Store Changes mode, see *Adding tasks (page 110)*.
- Kafka topic names cannot exceed 255 characters (249 from Kafka 0.10) and can only contain the following characters:
`a-z|A-Z|0-9|. (dot)|_(underscore)|-(minus)`
If the source table names exceed the maximum permitted length or contain unsupported characters, you need to either modify the names before starting the task or define a global transformation. For information on defining global transformations, see *Starting the Global Transformation Rules wizard (page 1107)*.
- The [Ignore ALTER](#) Apply Changes setting is not supported for changes to source data types and table renaming.
- Column names must begin with `[A-Za-z_]` (letters or an underscore) followed by `[A-Za-z0-9_]` (letters, digits, or an underscore). For example, `_Test_` is a valid column name whereas `&Test` is not.
If a source column name does not adhere to this rule, then [a transformation should be used](#) to rename the column.
- The [Drop and Create table](#) **Target Table Preparation** option is not supported
- The [Truncate before loading](#) **Target Table Preparation** option is not supported

- The DDL History and Change Data Partitioning [Control tables](#) are not supported
- DDL Changes: Dropping or renaming a table is not supported

Kafka target data types

The following table shows the default mapping from Qlik Replicate data types to Kafka data types.

For information on source data type mappings, see the section for the source endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.



When using the JSON message format, binary values are represented as hexadecimal digits.

Kafka data mapping

Qlik Replicate Data Types	Kafka Target Data Types in Schema Messages
DATE	DATE
TIME	TIME
DATETIME	DATETIME
BYTES	BYTES (length)
BLOB	BLOB
REAL4	REAL4 (7)
REAL8	REAL8 (14)
INT1	INT1 (3)
INT2	INT2 (5)
INT4	INT4 (10)
INT8	INT8 (19)
UINT1	UINT1 (3)
UINT2	UINT2 (5)
UINT4	UINT4 (10)
Values larger than $2^{31}-1$ are not supported.	
UINT8	UINT8 (20)
Values larger than $2^{63}-1$ are not supported.	
NUMERIC	NUMERIC (p,s)

Qlik Replicate Data Types	Kafka Target Data Types in Schema Messages
STRING	STRING (Length)
WSTRING	STRING (Length)
CLOB	CLOB
NCLOB	NCLOB
BOOLEAN	BOOLEAN (1)

Mapping to JSON and Avro

Avro message format uses logical types for a more precise representation of the data type.



Replicate data types will only be mapped to supported Avro logical data types if the [Use logical data types for specific data types](#) check box is selected.

Data type mappings

Qlik Replicate Data Types	JSON or Avro Primitive Data Types (When not using logical data types)	Avro Logical Data Types
DATE	STRING	DATE Annotates an Avro INT.
TIME	STRING	TIME-MILLIS Annotates an Avro INT.
TIMESTAMP	STRING	TIMESTAMP-MICROS Annotates an Avro LONG.
STRING	STRING	-
WSTRING	STRING	-
CLOB	STRING	-
NCLOB	STRING	-
NUMERIC	STRING	DECIMAL (p,s) Annotates an Avro BYTES.
BYTES	BYTES	-

Qlik Replicate Data Types	JSON or Avro Primitive Data Types (When not using logical data types)	Avro Logical Data Types
BLOB	BYTES	-
REAL4	FLOAT	-
REAL8	DOUBLE	-
INT1	INT	-
INT2	INT	-
INT4	INT	-
UINT1	INT	-
UINT2	INT	-
UINT4	LONG	-
INT8	LONG	-
UINT8	STRING	DECIMAL (20,0) Annotates an Avro BYTES.
BOOLEAN	BOOLEAN	-

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To define the general connection properties:

1. Click the **Manage Endpoint Connections** toolbar button.
The **Manage Endpoints Connections** dialog box opens.
2. Click the **New Endpoint Connection** toolbar button.
The **Name**, **Description**, **Type** and **Role** fields are displayed on the right.
3. In the **Name** field, specify a display name for the endpoint.
4. In the **Description** field, optionally type a description for the Kafka endpoint.
5. Select **Target** as the endpoint **Role**.
6. Select **Kafka** as the endpoint **Type**.
The dialog box is divided into **General** and **Advanced** tabs.
7. In the **Broker servers** field, specify one or more broker servers using the following format (for high availability):

```
server1[:port1] [, server2[:port2]]
```

Example:

```
192.168.1.100:9092,192.168.1.101:9093
```

Replicate will connect to the first available host. If a host is specified without a port then port 9092 will be used as the default.



When using SSL or Kerberos authentication, you must specify the broker FQDN (i.e. not the IP address).



*All of the broker servers in your cluster need to be accessible to Replicate. However, you do not need to specify all of the servers in the **Broker servers** field. This is because Replicate only need to connect to one of the servers in order to retrieve the connection details for the other servers in the cluster. It is therefore best practice to specify the servers that are most likely to be available when the task is run. The servers to which Replicate produces messages is determined by the topic and partitioning topic and partitioning settings described below.*

8. Continue as described below.

Security



- The **Use SSL** and **Certificate** authentication options are only supported from Kafka 0.9 or later.
- The CA file, public key file and private key file must all be in PEM format.
- The **Kerberos** and **User name and password** authentication methods are only supported from Kafka 0.10 or later.
- All of the broker servers in the cluster must be configured to accept connection requests using the selected **Authentication** method.

In the **Security** section, the following options can be set:

- **Use SSL** (supports TLS 1.0, 1.1 and 1.2): Select this option to encrypt the communication between the Replicate machine and the broker server(s). If the brokers are configured to require SSL, then you *must* select this option.
 - **CA path**: Specify either the full path (i.e. including the file name) to a specific CA certificate in PEM format or the directory containing certificate files with hash names.
- **Authentication**: Select one of the following:
 - **None** - No authentication.
 - **Certificate** - If you select this option, you also need to provide the following information:
 - Note** The public and private key files must be in PEM format.
 - **Public key file** - The full path to the public key file on the Replicate Server machine.
 - **Private key file** - The full path to the private key file on the Replicate Server machine.
 - **Private key password** - The password for the private key file.
 - **Kerberos (SASL/GSSAPI)** - Select to authenticate against the Kafka cluster using Kerberos. Replicate automatically detects whether Qlik Replicate Server is running on Linux or on Windows and displays the appropriate settings.

Qlik Replicate Server on Linux:

- **Principal** - The Kerberos principal used to authenticate against the broker server(s).
- **Keytab file** - The full path to the keytab file (that contains the specified principal) on the Replicate Server machine.



In order to use Kerberos authentication on Linux, the Kerberos client (workstation) package should be installed.

Qlik Replicate Server on Windows:



Note Both Replicate Server and the Kafka brokers must be connected to Active Directory KDC.

- **Realm** - The name of the domain in which the broker servers reside.
- **Principal** - The user name to use for authentication. The principal must be a member of the domain entered above.
- **Password** - The password for the principal entered above.

For additional steps required to complete setup for Kerberos authentication, see [Using Kerberos Authentication on Windows](#).

- **Username and password (SASL/PLAIN)** - You can select this option to authenticate yourself using a user name and password (SASL/PLAIN). To prevent the password from being sent in clear text, it is strongly recommended to enable the [Use SSL](#) option as well.
- **Username and Password (SASL/SCRAM-SHA-256)** - You can select this option to authenticate yourself using a user name and password (SASL/SCRAM-SHA-256). Note that selecting this option also requires each broker's `server.properties` file to be configured with the corresponding SASL/SCRAM mechanism.
- **Username and Password (SASL/SCRAM-SHA-512)** - You can select this option to authenticate yourself using a user name and password (SASL/SCRAM-SHA-512). Note that selecting this option also requires each broker's `server.properties` file to be configured with the corresponding SASL/SCRAM mechanism.

Message properties

In the **Message Properties** section, set the following properties:

1. Choose **JSON** or **Avro** as the message format.



Qlik provides an Avro Message Decoder SDK for consuming Avro messages produced by Qlik Replicate. You can download the SDK as follows:

1. Go to [Product Downloads](#).
2. Select **Qlik Data Integration**.
3. Scroll down the **Product** list and select **Replicate**.
4. In the **Download Link** column, locate the **QlikReplicate_<version>_Avro_Decoder_SDK.zip** file. Before starting the download, check the **Version** column to make sure that the version correlates with the Replicate version you have installed.
5. Proceed to download the **QlikReplicate_<version>_Avro_Decoder_SDK.zip** file.

For usage instructions, see *Kafka Avro consumers API* (page 1339).

An understanding of the Replicate envelope schema is a prerequisite for consuming Avro messages produced by Qlik Replicate. If you do not wish to use the SDK, see [The Qlik Envelope](#) for a description of the Replicate envelope schema.

2. From the **Compression** drop-down list, optionally select one of the available compression methods (**Snappy** or **gzip**). The default is **None**.
3. If you selected Avro, optionally select the **Use logical data types for specific data types** check box to map some of the number-based Qlik Replicate data types to Avro logical data types. When this option is *not* selected (the default), all Qlik Replicate data types will be mapped to Avro primitive data types. For more information on Qlik Replicate to Avro data type mapping, see [Mapping from Qlik Replicate Data Types to Avro](#).
4. If the message **Format** is set to **Avro**, **Publish** is set to **Publish data schemas to Confluent Schema Registry** or **Publish data schemas to Hortonworks Schema Registry** (see below), and the **Message Key** is *not* set to **None**, you can select the **Encode message key in Avro format** check box. When this option is *not* selected (the default), the message key will be in text format.



If you are using the Confluent JDBC Sink Connector to consume messages, this option must be enabled.

Data message publishing

In the **Data Message Publishing** section, set the following properties:

1. In the **Publish the data to** field, choose one of the following:
 - **Specific topic** - to publish the data to a single topic. Either type a topic name or use the browse button to select the desired topic.
 - **Specific topic for each table** - to publish the data to multiple topics corresponding to the source table names.
The target topic name consists of the source schema name and the source table name, separated by a period (e.g. "dbo.Employees"). The format of the target topic name is important as you will need to prepare these topics in advance.



If the topics do not exist, configure the brokers with `auto.create.topics.enable=true` to enable Replicate to create the topics during runtime. Otherwise, the task will fail.

- From the **Partition strategy** drop-down list, field, select either **Random** or **By message key**. If you select **Random**, each message will be written to a randomly selected partition. If you select **By message key**, messages will be written to partitions based on the selected **By message key** (described below).
- From the **Message key** drop-down list, field, select one of the following:



Note *If the message **Format** is set to Avro and the **Encode message key in Avro format** option is enabled, the message key will be an Avro record with an Avro schema.*

- **None** - To create messages without a message key.
- **Schema and table name** - For each message, the message key will contain a combination of schema and table name (e.g. "dbo+Employees").
When **By message key** is selected as the **Partition strategy**, messages consisting of the same schema and table name will be written to the same partition.
- **Primary key columns** - For each message, the message key will contain the value of the primary key column.
When **By message key** is selected as the **Partition strategy**, messages consisting of the same primary key value will be written to the same partition.

Metadata message publishing

In the **Metadata Message Publishing** section, specify whether or where to publish the message metadata.

From the **Publish** drop-down list, select one of the following options:

- **Do not publish metadata messages**
When this option is selected, only the data messages will be published. Additionally, the **Wrap data messages with the Replicate Envelope** option (enabled by default) will be displayed. This option is useful for organizations that wish to leverage the [Qlik Envelope](#) structure to process the data messages. If you do not require the additional information provided by the Attunity Envelope (e.g. due to existing message consumption processes), then disable this option.
- **Publish metadata messages to a dedicated metadata topic**
If you select this option, either type the **Topic name** or use the Browse button to select the desired topic. This option is required if the message format is set to **Avro** since Avro-formatted messages can only be opened using the Avro schema.
- **Publish data schemas to the Confluent Schema Registry**
If you select this option, you must also configure the **Schema Registry Connection Properties** described below.

- **Publish data schemas to the Hortonworks Schema Registry**

If you select this option, you must also configure the **Schema Registry Connection Properties** described below.



- *It is strongly recommended not to publish schema messages to the same topic as data messages.*
- *If the topics do not exist, configure the brokers with `auto.create.topics.enable=true` to enable Replicate to create the topics during runtime. Otherwise, the task will fail.*
- *The Confluent and Hortonworks Schema Registry options support Avro message format only.*

Schema Registry connection properties

- **Schema Registry servers:** Specify one or more Schema Registry servers using the following format (for high availability):

When publishing data schemas to the Confluent Schema Registry:

```
server1:port1[, server2[:port2]]
```

Example:

```
192.168.1.100:8081,192.168.1.101:8081
```

Replicate will connect to the first available host.

When publishing data schemas to the Hortonworks Schema Registry:

```
server1:port1[, server2[:port2]]
```

Example:

```
192.168.1.100:7788,192.168.1.101:7788
```

Replicate will connect to the first available host.

- **Use SSL** (supports TLS 1.0, 1.1 and 1.2): Select this option to encrypt the data between the Replicate machine and the Schema Registry server(s). If the servers re configured to require SSL, then you *must* select this option.

- **CA path:** Specify one of the following:

- The full path (i.e. including the file name) to a specific CA certificate in PEM format
- The directory containing certificate files with hash names

- **Authentication** - Select one of the following Schema Registry authentication options:

- **None** - No authentication.
- **Kerberos** - Select to authenticate using Kerberos.



- *This option is only supported when publishing data schemas to the Hortonworks Schema Registry and when Qlik Replicate Server is running on Linux.*
- *In order to use Kerberos authentication on Linux, the Kerberos client (workstation) package should be installed.*

10 Adding and managing target endpoints

- **Principal** - The Kerberos principal used to authenticate against the Schema Registry.
- **Keytab file** - The full path to the keytab file (that contains the specified principal) on the Replicate Server machine.
- **Certificate** - Select to authenticate using a certificate.



This option is only supported when publishing to the Confluent Schema Registry.

If you select this option, you also need to provide the following information:

- **Public key file** - The full path to the public key file on the Replicate Server machine.
- **Private key file** - The full path to the private key file on the Replicate Server machine.
- **Private key password** - The password for the private key file.
- **User name and password** - Select to authenticate with a user name and password.



This option is only supported when publishing to the Confluent Schema Registry.

- **Certificate + User name and password** - Select to authenticate using both a certificate and a user name and password.



This option requires Replicate 2023 Service Release 1 or later, and is only supported when publishing to the Confluent Schema Registry.

The **Certificate + User name and password** authentication method requires you to set each authentication method separately.

To do this, from the **Authentication** drop-down list:

- a. Select **Certificate** and enter the certificate information, as described in the **Certificate** option above.
 - b. Select **User name and password** and enter the username and password for the Schema Registry.
 - c. Select **Certificate + User name and password** and save the endpoint settings.
- **Use proxy server** - Select to publish to the Schema Registry via a proxy server.



This option is only supported when publishing to the Confluent Schema Registry.

- **Host name** - The host name of the proxy server.
- **Port** - The port via which to access the proxy server.
- **Scheme** - Select which protocol to use to access the server (**HTTP** or **HTTPS**).
- **SSL CA Path** - The location of the CA file on the Replicate Server machine when **HTTPS** is the selected **Scheme**.

Schema Registry subject properties

Subject Name Strategy



- The first strategy (Schema and Table Name Strategy) is a proprietary Qlik strategy while the other three are standard Confluent subject name strategies.
- For strategies with "Topic" in the subject name, the following should be considered:
 - When the "add \$topic column" method is used, the subject will be created only once (as the \$topic expression might create multiple subjects).
See also steps 3 and 4 in [Overriding the default settings](#) (page 834).
 - The "Metadata only" [Advanced run](#) option is not supported. This is because Replicate depends on the arrival of the first record per table in order to create the subject.

Select one of the available subject name strategies.

- Schema and Table Name - The default
- Topic Name
- Record Name
- Topic and Record Name - See also: [Message Format](#) (page 1196).

For more information on Confluent's subject name strategies, see

<https://docs.confluent.io/platform/current/schema-registry/serdes-develop/index.html#subject-name-strategy>

Subject Compatibility Mode

Select a compatibility mode from the **Subject compatibility mode** drop-down list. A description of the selected mode will appear below the drop-down list.



- Depending on the selected Subject Name Strategy, some of the compatibility modes may not be available.
- When [publishing messages to a Schema Registry](#), the default subject compatibility mode for all newly created Control Table subjects will be *None*, regardless of the selected **Subject compatibility mode**.
Should you wish the selected **Subject compatibility mode** to apply to Control Tables as well, set the `setNonCompatibilityForControlTables` internal parameter to `false`.

Overriding the default settings

A transformation can be defined that overrides the topic, partition and message key settings defined in the **General** tab.



Before you can define such a transformation, you first need to add a source endpoint to the task and select the tables you want to replicate.

To define a transformation:

1. Open the task you defined.
2. If you are defining a transformation for a single table, select one of the source tables. Otherwise, skip to Step 3.
3. Define a transformation that adds one of the following columns:



The columns listed below (prefixed with a \$) instruct Replicate to route the message to the desired topic and/or partition, and will not be included in the actual message itself.

- \$topic - To write messages to a specific topic.
- \$partition - To write messages to a specific partition.
- \$key - To create a custom message key.

For information on creating a transformation for a single table, see *Defining transformations for a single table/view* (page 1084).

For information on creating a global transformation rule, see *Starting the Global Transformation Rules wizard* (page 1107).

4. Define an expression for the new column that returns the following values:
 - For a \$topic column, the expression should return the topic name.
 - For a \$partition column, the expression should return the partition number. Note that an error will be returned during runtime if the partition number does not exist.
 - For a \$key column, the expression should return the message key contents. Note that the expression must return a non-empty value.

For information on creating expressions, see *Using the Expression Builder* (page 1137).

Setting advanced connection properties

In the **Advanced** tab, you can define advanced properties for the Kafka target endpoint:

- **Message Maximum Size**

In the **Message maximum size** field, specify the maximum size of messages that the broker(s) are configured to receive (`message.max.bytes`). Replicate will not send messages larger than the maximum size. Adjusting the maximum message size might improve performance in certain cases.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

Working with Confluent Cloud

The following topic explains how to configure the Replicate Kafka endpoint to work with Confluent Cloud.

1. Create a cluster in Confluent Cloud.
2. Copy the **Bootstrap server** value from the cluster settings to the **Broker servers** field in **General** tab of the Replicate Kafka endpoint settings.
3. In the **General** tab of the Replicate Kafka endpoint settings, select **Username and Password (SASL/PLAIN)** from the **Authentication** drop-down list.
4. In your Confluent Cluster's **API access** screen, create an **API key pair**.
5. Copy the key and the secret to the Kafka endpoint's **Username** and **Password** fields respectively.
6. In the Kafka endpoint settings, select the **Use SSL** option and specify the full path to the CA file in the **CA path** field. The CA file should be created according to Confluent's guidelines and should contains the CA that signs all of the Confluent Cloud certificates as well as its Root CA Signer.

You should be all set now to work with Confluent Cloud.

The Replicate envelope

All Replicate message types covered in this section are encapsulated in a single message schema called the Replicate envelope. The schema of the Replicate envelope is as following:

```
{
  "type": "record",
  "name": "MessageEnvelope",
  "fields": [
    {"name": "magic", "type": {"type": "fixed", "name": "Magic", "size": 5}},
    {"name": "type", "type": "string"},
    {"name": "headers", "type": [{"null"}, {"type": "map", "values": "string"}]},
    {"name": "messageSchemaId", "type": [{"null"}, "string"]},
    {"name": "messageSchema", "type": [{"null"}, "string"]},
    {"name": "message", "type": "bytes"}
  ]
}
```

The fields in the envelope are as follows:

- `magic` (5 bytes fixed field)
The constant "atMSG" is used to identify this form of message. The "atMSG" constant should be used to validate that this message is indeed an Replicate envelope message.
- `type` (string field)
Describes the enveloped message type. This can be one of two values: MD which stands for metadata message and DT which stands for data message.
- `headers` (map of string key and value)
A free for use map for various properties set at the application level. Currently, no headers are set by Qlik Replicate but this may change in future versions.
- `messageSchemaId` (null or string)
A reference to a schema defined elsewhere, which can be used to deserialize the bytes in the message field. This specification does not explain how the schema ID is used for looking up the actual schema - it is an application level detail. This field is used exclusively with the `messageSchema` field.
- `messageSchema` (null or string)
An embedded UTF-8 encoded Avro JSON schema with which the message field can be serialized. This field is used exclusively with the `messageSchemaId` field.
- `message` (bytes)
An Avro encoded message, which is the payload of the message envelope.

Given the envelope schema, it is possible for anyone using this schema to properly decode the envelope messages from Kafka.

Once the envelope message has been decoded, there are two possible scenarios:

- **Scenario 1:** Decoding a self-describing message such as the metadata message
- **Scenario 2:** Decoding a message by referenced schema ID such as data messages

The method for logically decoding messages in both scenarios is described below.

Decoding a self-describing message

When the `messageSchema` field is not null, it means the `message` field can be decoded using the schema included in the `messageSchema` field. This is fairly straightforward to perform programatically since the only thing you need to usually supply Avro is a schema and a message, both of which are provided in the envelope message.

The Replicate metadata messages which include both table metadata, lineage and data schema description (to be referenced later by data messages) are enveloped in the self-describing envelope.

Decoding a message by referenced schema ID

Avro schemas are JSON documents which can be quite large, usually much larger than the data encoded by Avro conforming to the schema. For example, a schema of a 10 column table could be a JSON document of more than 100 characters while an actual row encoding of 10 columns may be only 10 bytes (depending of course on the type and length of fields). It is therefore typically not recommended to include schema and data together in a Kafka message because the schema information is redundant and is the same for all data messages while the actual data is the only thing which differs between data messages.

10 Adding and managing target endpoints

To avoid sending schema with each data message, each schema has a 32 bytes long ID. When a data message based on a previously sent data message schema (via the metadata message) is constructed, the `messageSchema` field is set to null and the `messageSchemaId` field is set to the 32 bytes ID of the schema instead. The application responsibility is to locate the data schema sent earlier in the metadata message and use that schema to decode the data message contained in the `message` field.

Typical consumer logic

A typical scenario involving Kafka involves Qlik Replicate as the Producer of messages into Kafka and customer code as the Consumer. Qlik Replicate offers the ability to define a specific topic as the schema topic and different topics for the table data.

The customer's consumer code should read metadata messages from the schema topic and then save the data schemas and any other information the consumer wishes to access later in a customer defined zone. Another set of customer consumers should read data messages from the various data topics, and access the data schemas zone as required to retrieve the data schemas required for decoding the data messages.

When consuming data messages and metadata messages from several topics and partitions in a multi-thread/process manner, a situation may arise where a given consumer may attempt to read a data message *before* the corresponding metadata message has been read. As it is not possible to read a data message before its corresponding metadata message, the consumer's logic should wait a reasonable amount of time until the corresponding metadata message has been read. If the metadata message is still not available after waiting for a reasonable amount of time, the consumer should handle this as an unexpected error and activate the planned error policy. An example of such a policy could be saving the message in a dedicated "delayed" topic for later processing.

As a rule of thumb, the number of metadata messages will be much lower (in the magnitude of 1:10000 or more) than the number of data messages. So, assuming a metadata consumer is active, the gap between metadata message and data message should be no more than a few seconds (usually, milliseconds).

Metadata and data messages

This topic describes the structure and content of the metadata and data messages produced by the Kafka target endpoint.

Metadata message

Metadata message

Field	Type	Description
<code>schemald</code>	String	The unique identifier of the Avro schema.
<code>lineage</code>	Structure	Information about the origin of the data (Replicate server, task, table, and so on)
<code>server</code>	String	The name of the Replicate server.
<code>task</code>	String	The name of the task.
<code>schema (Schema Name)</code>	String	The name of the database schema.

10 Adding and managing target endpoints

Field	Type	Description
table (Table Name)	String	The name of the table.
tableVersion	Integer	Replicate maintains a version number of the structure of source table. Upon DDL change on the source, the version is increased and a new metadata message is produced.
timestamp	String	The date and time of the metadata message.
tableStructure	Structure	Describes the structure of the table.
tableColumns	Structure	Contains the list of columns and their properties.
{columns}	Structure	For each column, a record with the below properties.
ordinal	Integer	The position of the column in the record.
type	String	The column data type.
length	Integer	The maximum size of the data (in bytes) permitted for the column.
precision	Integer	For NUMERIC data type, the maximum number of digits required to represent the value.
scale	Integer	For NUMERIC data type, the maximum number of digits to the right of the decimal point permitted for a number.
primaryKeyPosition	Integer	The position of the column in the table's Primary Key. or Unique Index. The value is zero if the column is not part of the table's Primary Key.
dataSchema	String	The Avro schema for deserializing the Data messages.

Data message


Data message

Field	Type	Description
schema (name)	String	The name of the source database schema containing the replicated source table(s).
table (name)	String	The name of the source table.
headers	Structure	Information about the current record.


10 Adding and managing target endpoints

Field	Type	Description
operation (Operation)	Enum	The operation type. Full Load - Initial load of the source data to the target topic(s) REFRESH - Insertion of a record to the target during Full Load CDC - Applies source table changes to the relevant topic INSERT - Insertion of new target record UPDATE - Update of an existing target record DELETE - Deletion of a target record
changeSequence (Change Sequence)	String	A monotonically increasing change sequencer that is common to all change tables of a task. Use this field to order the records in chronological order. Applicable to CDC operations only.
timestamp (Timestamp)	String	The original change UTC timestamp. Applicable to CDC operations only.
streamPosition	String	The source CDC stream position. Applicable to CDC operations only.
transactionId (Transaction ID)	String	The ID of the transaction that the change record belongs to. Use this field to gather all changes of a specific transaction. Applicable to CDC operations only.

10 Adding and managing target endpoints

Field	Type	Description
changeMask (Change Mask)	String	<p>Indicates which data columns were changed in the source table.</p> <p>The change mask is a string of hexadecimal digits, representing a bitmask of data columns in little-endian order. The bit position in the change mask is based on the ordinal of the column in the metadata message of that table.</p> <p>This means that if there are 10 data columns, they occupy bits 0 to 9 in the bitmask.</p> <p>If UPDATE mask is 0B hexadecimal, which is 1011 binary – it means that the columns at ordinals 1, 2 and 4 were changed.</p> <p>The following describes the bit semantics:</p> <ul style="list-style-type: none">• For INSERT records, all the inserted columns have the associated bits set.• For DELETE records, only primary-key (or unique index) columns have the associated bits set. This allows an applier to construct a DELETE statement without having to find the primary key fields from another source.• For UPDATE records, each column with a changed value will have the associated bit set. <div data-bbox="699 1205 1390 1339" style="border: 1px solid #ccc; padding: 5px;"><p> <i>LOB columns are not included in the changeMask bit.</i></p></div>

10 Adding and managing target endpoints

Field	Type	Description
columnMask (Column Mask)	String	<p>Indicates which data columns are present in the message. Usually, this will include all of the table columns.</p> <div style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p> <i>When replicating from an Oracle source without full supplemental logging, some columns might not be present in the data, since they could not be replicated.</i></p> </div> <p>The column mask is a string of hexadecimal digits, representing a bitmask of data columns in little-endian order. The bit position in the column mask is based on the ordinal of the column in the metadata message for that table.</p> <p>This allows the applier to distinguish a null value that is the <i>actual</i> value of the column, from a null value that represents a column which could not be replicated from the source database.</p>
externalSchemald	String	<p>The Schema ID. This will only be displayed if the Include external Schema ID header check box is selected.</p> <p>As the Schema ID changes whenever a DDL is performed on the source table, consumer applications can use this information to determine if the message schema has changed.</p>
transactionEventCounter (Transaction Event Counter)	Long	<p>The sequence number of the current operation in the transaction.</p> <p>This can be used to determine the order of operations within a transaction.</p>
transactionLastEvent (Transaction Last Event)	Boolean	<p>"True" indicates that it is the final record in the transaction whereas "False" indicates that not all of the records have been processed.</p>
data	Structure	The data of the table record
{columns}		The column names and values in the current record.
beforeData	Structure	The data of the table record, before the change
{columns}		<p>The column names and values, before the change.</p> <p>Applicable to UPDATE operation.</p>

Using Kerberos authentication on Windows

Whether Qlik Replicate Server is running on Linux or Windows, you can configure it to authenticate itself against the Kafka cluster using Kerberos.

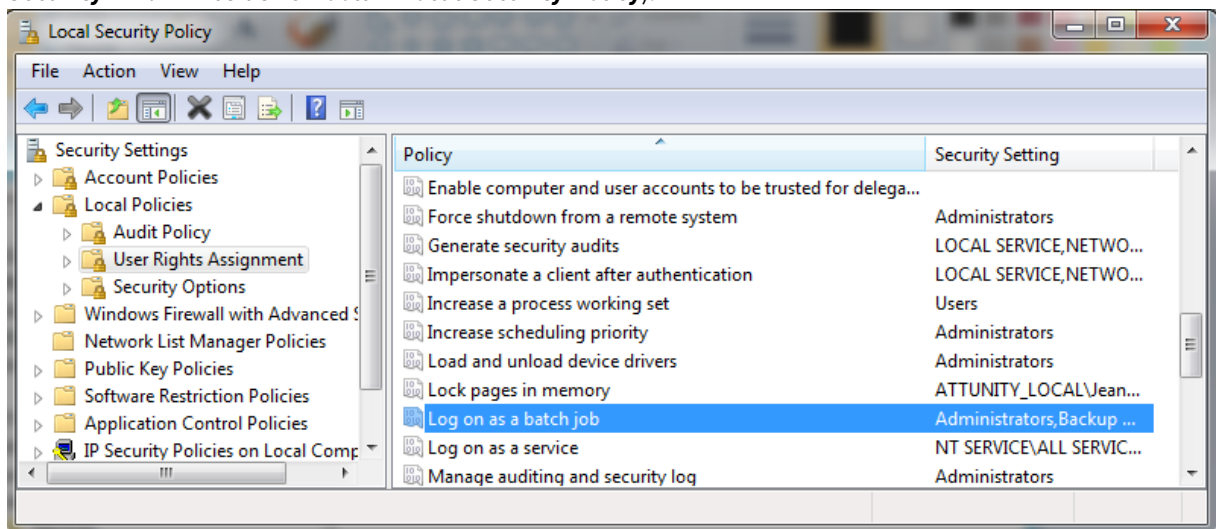
10 Adding and managing target endpoints

On Windows, this requires you to perform the following steps on the Qlik Replicate machine before starting the Qlik Replicate Server.

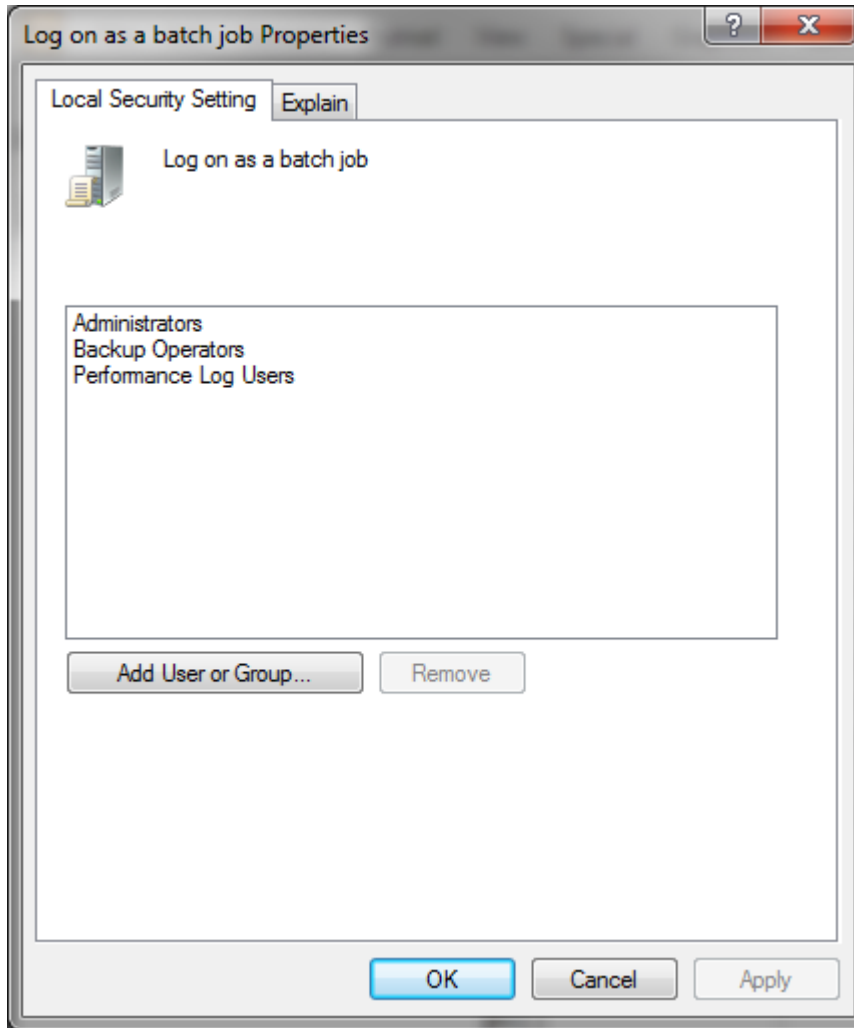
Before beginning, make sure that the impersonated user (principal) is granted read and write permissions on the Replicate **Data** directory (<product_dir>\Data by default) on the Qlik Replicate server. For Active Directory KDC, the impersonated user is the user configured in the user interface.

To set up Kerberos authentication on Windows, perform the following steps to ensure that the impersonated user (principal) has the **Log on as a batch job** privilege on the Qlik Replicate server.

1. On the Qlik Replicate Server machine, open the **Local Security Settings (Control Panel > System Security > Administrative Tools > Local Security Policy)**.



2. In the console tree, expand **Local Policies** and select **User Rights Assignments**.
3. In the details pane, double-click **Log on as a batch job**.
4. In the **Log on as a batch job Properties** dialog box, on the **Local Security Settings** tab, verify that the relevant user is listed. If the user is not listed, click **Add User or Group**, then add the user and click **OK**.



Your changes should take effect immediately.

10.24 Using a Log Stream target

For information on using Log Stream as a target, see *Using the Log Stream* (page 1072).

10.25 Using MapR Streams as a target

This section describes how to set up and use MapR Streams as a target endpoint in a replication task. In a task with a MapR Streams target endpoint, each source record is transformed into a message which is then written (with an optional message key) to a partition in the specified topic.

In this section:

- *Transaction processing by the consumer (page 848)*
- *Prerequisites (page 849)*
- *Limitations (page 849)*
- *Supported data types (page 850)*
- *Setting general connection properties (page 852)*
- *Setting advanced connection properties (page 855)*
- *The Replicate envelope (page 855)*
- *Metadata and data messages (page 857)*

Transaction processing by the consumer

When configuring the Qlik Replicate MapR Streams endpoint, users can configure various settings that affect where messages are published within the MapR Streams infrastructures (topics/partitions).

During a task's CDC stage, committed changes that are detected by the Qlik Replicate source endpoint are grouped by transaction, sorted internally in chronological order, and then propagated to the target endpoint. The target endpoint can handle the changes in various ways such as applying them to the target tables or storing them in dedicated Change Tables.

Each CDC message has both a transaction ID as well as change sequence. As the change sequence is a monotonically growing number, sorting events by change sequence always achieves chronological order. Grouping the sorted events by transaction ID then results in transactions containing chronologically sorted changes.

However, as MapR Streams is a messaging infrastructure, applying changes is not feasible while storing changes in tables is meaningless. The Replicate MapR Streams endpoint, therefore, takes a different approach, which is to report all transactional events as messages.

How it works

Each change in the source system is translated to a data message containing the details of the change including the transaction ID and change sequence in the source. The data message also includes the changed columns before and after the change. As explained above, the order in which the MapR Streams target writes the messages is the same as order of changes within each transaction.

Once a data message is ready to be sent to MapR Streams, the topic and partition it should go to are determined by analyzing the endpoint settings and any transformation settings. For example, the user might decide to configure the endpoint in such a way that every table is sent to a different topic and set the partition strategy to "Random", meaning that each message (within the same table) will be sent to a different partition.

Transaction consistency from a consumer perspective

If maintaining transaction consistency is important for the consumer implementation, it means that although the transaction ID exists in all data messages, the challenge is to gather the messages in a way that would facilitate identifying a whole transaction. An additional challenge is getting the transaction in the original order they were committed, which could be an even greater challenge if transactions are spread across multiple topics and partitions.

The simplest way of achieving the above goal is to direct Replicate to a specific topic and a specific partition (in the endpoint settings). This means that all data messages will end up in a single partition, thus guaranteeing ordered delivery both of transactions *and* of changes within a transaction. The consuming application could then consume messages - accumulating a transaction in some intermediate memory buffer - and when a new transaction ID is detected, mark the previous transaction as completed.

Although the simple way may work, it's not very efficient at the task level as all messages end up in the same topic and partition, not necessarily utilizing the full parallelism of the MapR Streams cluster. This may be a non-issue if there are multiple tasks, each taking advantage of a different topic/partition. In such as scenario, the gathering of messages from those tasks may very well utilize the cluster optimally.

The more generic way where data may be spread over multiple topics and partitions means that some intermediate buffer such as memory, a table in a relational database, or even other topics would need to be used to collect information about transactions. Then, the transactions would need to be rebuilt by periodically (every few minutes/hours) sorting the events collected from Replicate's MapR Streams output by the change sequence and grouping them by transaction ID.

Prerequisites

Before you can use MapR Streams as a target endpoint in a Replicate task, the following prerequisites must be met:

- Install and configure MapR Client 5.2.1 or later on the Qlik Replicate Server machine.
To verify the configuration, issue the following command in a Linux shell:

```
hadoop fs -ls maprfs://<cluster_name>/
```


The command should return a directory listing.
- Install the **mapr-librdkafka** package on the Qlik Replicate Server machine.
- Create a topic named **attrep_apply_exceptions** before starting the replication task. Note that if this topic does not exist, the task will always fail when it encounters a data error, regardless of the [error handling policy](#).
For a description of the **attrep_apply_exceptions** table, see *Apply exceptions (page 1298)*
- Add the following files to the `/opt/attunity/replicate/lib` directory:
 - `librdkafka.so`, `libMapRClient_c.so`, and `libMapRClient.so` (copy from the `/opt/mapr/lib` directory)
 - `libjvm.so` (copy from the `$JAVA_HOME/lib/amd64/server/lib` directory)

Limitations

When defining a task with MapR Streams as the target endpoint, the following limitations apply:

- Unlimited LOB size is not supported. Therefore, when replicating from source tables with LOB columns, *do not* select the **Allow unlimited LOB size** option.
For more information on defining LOB settings, see *Target metadata (page 1166)*.
- **Batch optimized apply** mode is not supported. If this mode is set, the task will automatically switch to **Transactional apply** mode and issue an appropriate warning.
For more information on these modes, see *Change Processing Tuning (page 1181)*.
- Store Changes mode is not supported.
For more information on Store Changes mode, see *Adding tasks (page 110)*.
- MapR Streams topic names cannot exceed 255 characters and can only contain the following characters:
`a-z|A-Z|0-9|. (dot)|_(underscore)|-(minus)`
If the source table names exceed the maximum permitted length or contain unsupported characters, you need to either modify the names before starting the task or define a global transformation. For information on defining global transformations, see *Starting the Global Transformation Rules wizard (page 1107)*.
- The [Ignore ALTER](#) Apply Changes setting is not supported for changes to source data types and table renaming.

10 Adding and managing target endpoints

- Column names must begin with [A-Za-z_] (letters or an underscore) followed by [A-Za-z0-9_] (letters, digits, or an underscore). For example, **_Test_** is a valid column name whereas **&Test** is not. If a source column name does not adhere to this rule, then [a transformation should be used](#) to rename the column.

Supported data types

The following table shows the default mapping from Qlik Replicate data types to MapR Streams data types.

For information on source data type mappings, see the section for the source endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).




When using the JSON message format, binary values are represented as hexadecimal digits.

Supported data types

Qlik Replicate Data Types	MapR Streams Target Data Types in Schema Messages
DATE	DATE
TIME	TIME
DATETIME	DATETIME
BYTES	BYTES (length)
BLOB	BLOB
REAL4	REAL4 (7)
REAL8	REAL8 (14)
INT1	INT1 (3)
INT2	INT2 (5)
INT4	INT4 (10)
INT8	INT8 (19)
UINT1	UINT1 (3)
UINT2	UINT2 (5)
UINT4	UINT4 (10)

Values larger than $2^{31}-1$ are not supported.

Qlik Replicate Data Types	MapR Streams Target Data Types in Schema Messages
UINT8 <div style="border: 1px solid black; padding: 5px; margin-top: 10px;">  Values larger than $2^{63}-1$ are not supported. </div>	UINT8 (20)
NUMERIC	NUMERIC (p,s)
STRING	STRING (Length)
WSTRING	STRING (Length)
CLOB	CLOB
NCLOB	NCLOB
BOOLEAN	BOOLEAN (1)

Mapping to JSON and Avro

Avro message format uses logical types for a more precise representation of the data type.

Data type mappings

Qlik Replicate Data Types	JSON or Avro Primitive Data Types	Avro Logical Data Types
DATE	STRING	DATE Annotates an Avro INT.
TIME	STRING	TIME-MILLIS Annotates an Avro INT.
TIMESTAMP	STRING	TIMESTAMP-MICROS Annotates an Avro LONG.
STRING	STRING	-STRING
WSTRING	STRING	-STRING
CLOB	STRING	-STRING
NCLOB	STRING	-STRING
NUMERIC	STRING	DECIMAL (p,s) Annotates an Avro BYTES.
BYTES	BYTES	-BYTES
BLOB	BYTES	-BYTES
REAL4	FLOAT	-FLOAT

Qlik Replicate Data Types	JSON or Avro Primitive Data Types	Avro Logical Data Types
REAL8	DOUBLE	-DOUBLE
INT1	INT	-INT
INT2	INT	-INT
INT4	INT	-INT
UINT1	INT	-INT
UINT2	INT	-LONG
UINT4	LONG	-LONG
INT8	LONG	-LONG
UINT8	STRING	DECIMAL (20,0) Annotates an Avro BYTES.
BOOLEAN	BOOLEAN	-BOOLEAN

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To define the general connection properties:

1. Click the **Manage Endpoint Connections** toolbar button.
The **Manage Endpoints Connections** dialog box opens.
2. Click the **New Endpoint Connection** toolbar button.
The **Name**, **Description**, **Type** and **Role** fields are displayed on the right.
3. In the **Name** field, specify a display name for the endpoint.
4. In the **Description** field, optionally type a description for the MapR Streams endpoint.
5. Select **Target** as the endpoint **Role**.
6. Select **MapR Streams** as the endpoint **Type**.
The dialog box is divided into **General** and **Advanced** tabs.
7. In the **Cluster name** field, specify the MapR Streams cluster name.
To check if the cluster is valid, execute the following command in a Linux shell:
`hadoop fs -ls maprfs://cluster_name/`
If there is a directory listing for the cluster, the cluster is valid.
8. In the **Data Publishing** section, set the following properties:
 - a. In the **Publish the data to** field, choose one of the following:
 - **Specific topic** - to publish the data to a single topic. Type the stream name followed by a colon and the topic name.
Example:
`MapRStreamName:MapRTopicName`
 - **Specific topic for each table** - to publish the data to multiple topics corresponding to the source table names.

10 Adding and managing target endpoints

In the **Stream name** field, enter the stream name only (i.e. *stream-name* as opposed to *stream name:topic-name*).



If the topics do not exist, the stream must be configured with `autocreate=true` to enable Replicate to create the topics during runtime. Otherwise, the task will fail.

- b. From the **Partition strategy** drop-down list, field, select either **Random** or **By message key**. If you select **Random**, each message will be written to a randomly selected partition. If you select **By message key**, messages will be written to partitions based on the selected **By message key** (described below).
- c. From the **Message key** drop-down list, field, select one of the following:



Note *The message key is represented as a string, regardless of the selected data message format (JSON/Avro).*

- **None** - To create messages without a message key.
- **Schema and table name** - For each message, the message key will contain a combination of schema and table name (e.g. "dbo+Employees").
When **By message key** is selected as the **Partition strategy**, messages consisting of the same schema and table name will be written to the same partition.
- **Primary key columns** - For each message, the message key will contain the value of the primary key column.
When **By message key** is selected as the **Partition strategy**, messages consisting of the same primary key value will be written to the same partition.

10. In the **Message Properties** section, set the following properties:

- a. Choose **JSON** or **Avro** as the message format.



Qlik provides an Avro Message Decoder SDK for consuming Avro messages produced by Qlik Replicate. You can download the SDK as follows:

1. Go to [Product Downloads](#).
2. Select **Qlik Data Integration**.
3. Scroll down the **Product** list and select **Replicate**.
4. In the **Download Link** column, locate the **QlikReplicate_<version>_Avro_Decoder_SDK.zip** file. Before starting the download, check the **Version** column to make sure that the version correlates with the Replicate version you have installed.
5. Proceed to download the **QlikReplicate_<version>_Avro_Decoder_SDK.zip** file.

For usage instructions, see [Kafka Avro consumers API](#) (page 1339).

An understanding of the Qlik envelope schema is a prerequisite for consuming Avro messages produced by Qlik Replicate. If you do not wish to use the SDK, see [The Qlik Envelope](#) for a description of the Qlik envelope schema.

- b. To publish the schema message (for the corresponding data message) to a topic, select the **Use Schema Messages** check box and then type the stream name followed by a colon and the topic name.

Example:

MapRStreamName:MapRTopicName

This option is required if the message format is set to **Avro** since Avro-formatted messages can only be opened using the Avro schema.



It is strongly recommended not to publish schema messages to the same topic as data messages.



If the topics do not exist, the stream must be configured with `autocreate=true` to enable Replicate to create the topics during runtime. Otherwise, the task will fail.

Overriding the default settings

A transformation can be defined that overrides the topic, partition and ordering key settings defined in the **General** tab.



Before you can define such a transformation, you first need to add a source endpoint to the task and select the tables you want to replicate.

To define a transformation:

1. Open the task you defined.
2. If you are defining a transformation for a single table, select one of the source tables. Otherwise, skip to Step 3.
3. Define a transformation that adds one of the following columns:



The columns listed below (prefixed with a \$) instruct Replicate to route the message to the desired topic and/or partition, and will not be included in the actual message itself.

- \$topic - To write messages to a specific topic.
- \$partition - To write messages to a specific partition.
- \$key - To create a custom ordering key.

For information on creating a transformation for a single table, see *Defining transformations for a single table/view* (page 1084).

For information on creating a global transformation rule, see *Starting the Global Transformation Rules wizard* (page 1107).

4. Define an expression for the new column that returns the following values:
 - For a `$topic` column, the expression should return the topic name.
 - For a `$partition` column, the expression should return the partition number. Note that an error will be returned during runtime if the partition number does not exist.
 - For a `$key` column, the expression should return the ordering key contents. Note that the expression must return a non-empty value.

For information on creating expressions, see *Using the Expression Builder (page 1137)*.

Setting advanced connection properties

In the **Advanced** tab, you can define advanced properties for the Mapr Streams target endpoint:

- **Message Maximum Size**
In the **Message maximum size** field, specify the maximum size of messages. Replicate will not send messages larger than the maximum size.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

The Replicate envelope

All Replicate message types covered in this section are encapsulated in a single message schema called the Replicate envelope. The schema of the Replicate envelope is as following:

```
{  
  "type": "record",  
  "name": "MessageEnvelope",
```

```
"fields": [
  {"name": "magic", "type": {"type": "fixed", "name": "Magic", "size": 5}},
  {"name": "type", "type": "string"},
  {"name": "headers", "type": ["null", {"type": "map", "values": "string"}]},
  {"name": "messageSchemaId", "type": ["null", "string"]},
  {"name": "messageSchema", "type": ["null", "string"]},
  {"name": "message", "type": "bytes"}
]
```

The fields in the envelope are as follows:

- **magic** (5 bytes fixed field)
The constant "atMSG" is used to identify this form of message. The "atMSG" constant should be used to validate that this message is indeed an Replicate envelope message.
- **type** (string field)
Describes the enveloped message type. This can be one of two values: MD which stands for metadata message and DT which stands for data message.
- **headers** (map of string key and value)
A free for use map for various properties set at the application level. Currently, no headers are set by Qlik Replicate but this may change in future versions.
- **messageSchemaId** (null or string)
A reference to a schema defined elsewhere, which can be used to deserialize the bytes in the message field. This specification does not explain how the schema ID is used for looking up the actual schema - it is an application level detail. This field is used exclusively with the messageSchema field.
- **messageSchema** (null or string)
An embedded UTF-8 encoded Avro JSON schema with which the message field can be serialized. This field is used exclusively with the messageSchemaId field.
- **message** (bytes)
An Avro encoded message, which is the payload of the message envelope.

Given the envelope schema, it is possible for anyone using this schema to properly decode the envelope messages from MapR Streams.

Once the envelope message has been decoded, there are two possible scenarios:

- **Scenario 1:** Decoding a self-describing message such as the metadata message
- **Scenario 2:** Decoding a message by referenced schema ID such as data messages

The method for logically decoding messages in both scenarios is described below.

Decoding a self-describing message

When the messageSchema field is not null, it means the message field can be decoded using the schema included in the messageSchema field. This is fairly straightforward to perform programmatically since the only thing you need to usually supply Avro is a schema and a message, both of which are provided in the envelope message.

The Replicate metadata messages which include both table metadata, lineage and data schema description (to be referenced later by data messages) are enveloped in the self-describing envelope.

Decoding a message by referenced schema ID

Avro schemas are JSON documents which can be quite large, usually much larger than the data encoded by Avro conforming to the schema. For example, a schema of a 10 column table could be a JSON document of more than 100 characters while an actual row encoding of 10 columns may be only 10 bytes (depending of course on the type and length of fields). It is therefore typically not recommended to include schema and data together in a MapR Streams message because the schema information is redundant and is the same for all data messages while the actual data is the only thing which differs between data messages.

To avoid sending schema with each data message, each schema has a 32 bytes long ID. When a data message based on a previously sent data message schema (via the metadata message) is constructed, the `messageSchema` field is set to null and the `messageSchemaId` field is set to the 32 bytes ID of the schema instead. The application responsibility is to locate the data schema sent earlier in the metadata message and use that schema to decode the data message contained in the `message` field.

Typical consumer logic

A typical scenario involving MapR Streams involves Qlik Replicate as the Producer of messages into MapR Streams and customer code as the Consumer. Qlik Replicate offers the ability to define a specific topic as the schema topic and different topics for the table data.

The customer's consumer code should read metadata messages from the schema topic and then save the data schemas and any other information the consumer wishes to access later in a customer defined zone. Another set of customer consumers should read data messages from the various data topics, and access the data schemas zone as required to retrieve the data schemas required for decoding the data messages.

When consuming data messages and metadata messages from several topics and partitions in a multi-thread/process manner, a situation may arise where a given consumer may attempt to read a data message *before* the corresponding metadata message has been read. As it is not possible to read a data message before its corresponding metadata message, the consumer's logic should wait a reasonable amount of time until the corresponding metadata message has been read. If the metadata message is still not available after waiting for a reasonable amount of time, the consumer should handle this as an unexpected error and activate the planned error policy. An example of such a policy could be saving the message in a dedicated "delayed" topic for later processing.

As a rule of thumb, the number of metadata messages will be much lower (in the magnitude of 1:10000 or more) than the number of data messages. So, assuming a metadata consumer is active, the gap between metadata message and data message should be no more than a few seconds (usually, milliseconds).

Metadata and data messages

This topic describes the structure and content of the metadata and data messages produced by the MapR Streams target endpoint.

Metadata message

Metadata message

Field	Type	Description
<code>schemald</code>	String	The unique identifier of the Avro schema.

10 Adding and managing target endpoints

Field	Type	Description
lineage	Structure	Information about the origin of the data (Replicate server, task, table, and so on)
server	String	The name of the Replicate server.
task	String	The name of the task.
schema (Schema Name)	String	The name of the database schema.
table (Table Name)	String	The name of the table.
tableVersion	Integer	Replicate maintains a version number of the structure of source table. Upon DDL change on the source, the version is increased and a new metadata message is produced.
timestamp	String	The date and time of the metadata message.
tableStructure	Structure	Describes the structure of the table.
tableColumns	Structure	Contains the list of columns and their properties.
{columns}	Structure	For each column, a record with the below properties.
ordinal	Integer	The position of the column in the record.
type	String	The column data type.
length	Integer	The maximum size of the data (in bytes) permitted for the column.
precision	Integer	For NUMERIC data type, the maximum number of digits required to represent the value.
scale	Integer	For NUMERIC data type, the maximum number of digits to the right of the decimal point permitted for a number.
primaryKeyPosition	Integer	The position of the column in the table's Primary Key. or Unique Index. The value is zero if the column is not part of the table's Primary Key.
dataSchema	String	The Avro schema for deserializing the Data messages.

Data message


Data message

Field	Type	Description
schema (name)	String	The name of the source database schema containing the replicated source table(s).
table (name)	String	The name of the source table.
headers	Structure	Information about the current record.


10 Adding and managing target endpoints

Field	Type	Description
operation (Operation)	Enum	The operation type. Full Load - Initial load of the source data to the target topic(s) REFRESH - Insertion of a record to the target during Full Load CDC - Applies source table changes to the relevant topic INSERT - Insertion of new target record UPDATE - Update of an existing target record DELETE - Deletion of a target record
changeSequence (Change Sequence)	String	A monotonically increasing change sequencer that is common to all change tables of a task. Use this field to order the records in chronological order. Applicable to CDC operations only.
timestamp (Timestamp)	String	The original change UTC timestamp. Applicable to CDC operations only.
streamPosition	String	The source CDC stream position. Applicable to CDC operations only.
transactionId (Transaction ID)	String	The ID of the transaction that the change record belongs to. Use this field to gather all changes of a specific transaction. Applicable to CDC operations only.

10 Adding and managing target endpoints

Field	Type	Description
changeMask (Change Mask)	String	<p>Indicates which data columns were changed in the source table.</p> <p>The change mask is a string of hexadecimal digits, representing a bitmask of data columns in little-endian order. The bit position in the change mask is based on the ordinal of the column in the metadata message of that table.</p> <p>This means that if there are 10 data columns, they occupy bits 0 to 9 in the bitmask.</p> <p>If UPDATE mask is 0B hexadecimal, which is 1011 binary – it means that the columns at ordinals 1, 2 and 4 were changed.</p> <p>The following describes the bit semantics:</p> <ul style="list-style-type: none">• For INSERT records, all the inserted columns have the associated bits set.• For DELETE records, only primary-key (or unique index) columns have the associated bits set. This allows an applier to construct a DELETE statement without having to find the primary key fields from another source.• For UPDATE records, each column with a changed value will have the associated bit set. <div data-bbox="702 1205 1388 1339" style="border: 1px solid #ccc; padding: 5px;"><p> <i>LOB columns are not included in the changeMask bit.</i></p></div>

10 Adding and managing target endpoints

Field	Type	Description
columnMask (Column Mask)	String	<p>Indicates which data columns are present in the message. Usually, this will include all of the table columns.</p> <div style="border: 1px solid gray; padding: 5px; margin: 10px 0;">  <i>When replicating from an Oracle source without full supplemental logging, some columns might not be present in the data, since they could not be replicated.</i> </div> <p>The column mask is a string of hexadecimal digits, representing a bitmask of data columns in little-endian order. The bit position in the column mask is based on the ordinal of the column in the metadata message for that table.</p> <p>This allows the applier to distinguish a null value that is the <i>actual</i> value of the column, from a null value that represents a column which could not be replicated from the source database.</p>
externalSchemald	String	<p>The Schema ID. This will only be displayed if the Include external Schema ID header check box is selected.</p> <p>As the Schema ID changes whenever a DDL is performed on the source table, consumer applications can use this information to determine if the message schema has changed.</p>
transactionEventCounter (Transaction Event Counter)	Long	<p>The sequence number of the current operation in the transaction.</p> <p>This can be used to determine the order of operations within a transaction.</p>
transactionLastEvent (Transaction Last Event)	Boolean	<p>"True" indicates that it is the final record in the transaction whereas "False" indicates that not all of the records have been processed.</p>
data	Structure	The data of the table record
{columns}		The column names and values in the current record.
beforeData	Structure	The data of the table record, before the change
{columns}		<p>The column names and values, before the change.</p> <p>Applicable to UPDATE operation.</p>

10.26 Using Microsoft APS PDW as a target

This section describes how to set up and use Microsoft APS PDW as a target database in a replication task.

In this section:

- *Prerequisites (page 863)*
- *Limitations and considerations (page 863)*
- *Security requirements (page 863)*
- *Microsoft APS PDW target data types (page 863)*
- *Setting general connection properties (page 864)*
- *Setting advanced connection properties (page 865)*

Prerequisites



- Qlik Replicate must be installed on any Windows computer in your network.
- A Microsoft APS PDW account with the required access privileges is required.

The following client components must be installed on the Qlik Replicate machine:

- SQL Server Native Client 11.0
- Microsoft SQL Server 2012 Parallel Data Warehouse Tools x64

Limitations and considerations

Source columns with CHAR/VARCHAR data types and a non-Latin collation (e.g. "Chinese_PRC_CI_AS") need to be mapped to NVARCHAR. This can be done by defining a global transformation for all tables in the replication task or by defining a single transformation for a specific table.

For more information on defining transformations, see *Starting the Global Transformation Rules wizard (page 1107)* and *Defining transformations for a single table/view (page 1084)*.

Security requirements

You must provide Microsoft APS PDW account access to the Qlik Replicate user. This user must have LOAD permission and applicable permissions (INSERT, UPDATE, DELETE) on the destination table.

Microsoft APS PDW target data types

The Microsoft APS PDW database for Qlik Replicate supports most Microsoft APS PDW data types. The following table shows the Microsoft APS PDW target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source database you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

Target data types

Qlik Replicate Data Types	Microsoft APS PDW Data Types
BOOLEAN	BIT
BYTES	VARBINARY (Length)
DATE	DATE
TIME	TIME
DATETIME	DATETIME2 (scale)
INT1	TINYINT
INT2	SMALLINT

Qlik Replicate Data Types	Microsoft APS PDW Data Types
INT4	INTEGER
INT8	BIGINT
NUMERIC	DECIMAL (p,s)
REAL4	FLOAT (24)
REAL8	FLOAT (53)
STRING	VARCHAR (Length)
UINT1	TINYINT
UINT2	SMALLINT
UINT4	INTEGER
UINT8	BIGINT
WSTRING	NVARCHAR (Length)
BLOB	VARBINARY (8000)
NCLOB	NVARCHAR (4000)
CLOB	VARCHAR (8000)

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a Microsoft APS PDW target endpoint to Qlik Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Microsoft APS PDW database. This is optional.
4. Select **Target** as the database **role**.
5. Select **Microsoft APS PDW** as the database **Type**.
6. In the **Server name** field, enter the hostname or IP address of the Microsoft APS PDW machine.
7. Optionally, change the default **Port** (5000).
8. Type the Microsoft APS PDW authentication information (**User Name**, **Password**) for the authorized user for this Microsoft APS PDW database. If you do not know this information, see your Microsoft APS PDW database Administrator (DBA).



- *This information is required. If you are using the **Advanced** tab to create a custom string, make sure to include the **User Name** and **Password** properties. See *Setting advanced connection properties* (page 865) for more information.*
- *This information is case sensitive.*
- *If you want to set custom properties for this database, see *Setting advanced connection properties* (page 865).*



*Make sure that the Microsoft APS PDW user entered in the Microsoft APS PDW Authentication section has the correct access privileges. For information on how to provide the required privileges, see *Security requirements* (page 863).*

9. In the **Database name** field, enter the name of the Microsoft APS PDW database.

Setting advanced connection properties

In the **Advanced** tab, you can set the following parameters:

- **Maximum file size:** Select or type the maximum size (in KB) of a CSV file before the file is loaded into the Microsoft APS PDW database. The default value is 32000 KB.
- **Create hash distribution:** Enabling this option will turn on the Microsoft APS PDW hash distribution function.
- **Additional ODBC connection properties:** Specify any additional ODBC connection parameters that you want to use.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.27 Using Microsoft Azure ADLS as a target

This chapter describes how to set up and use Microsoft Azure ADLS as a target in a replication task.

In this section:

- *Prerequisites (page 868)*
- *Microsoft Azure ADLS target overview (page 868)*
- *Limitations and considerations (page 868)*
- *Change data partitioning (page 869)*
- *Data types (page 870)*
- *Setting general connection properties (page 871)*
- *Setting advanced connection properties (page 876)*
- *Generating reference files (page 879)*

Microsoft Azure ADLS target overview

When using Microsoft Azure ADLS as a target in a Replicate task, both the Full Load and Change Processing data are written to data files. Depending on the endpoint settings, the data file format can be CSV, JSON or Parquet. While the explanations in this section relate to CSV files, the same is true for JSON and Parquet files.

Full Load files are named using incremental counters such as LOAD00001.csv, LOAD 00002.csv, and so on, whereas Apply Changes files are named using timestamps e.g. 20141029-1134010000.csv.



When the [Create metadata files in the target folder](#) option is enabled, a corresponding metadata file is created using the same naming format, but with a .dfm extension.

For each source table, a folder is created in the specified Microsoft Azure ADLS target folder. The data files are created on the Replicate Server machine and are then uploaded to the specified Microsoft Azure ADLS target folder once the [File Attributes](#) (Full Load) and [Change Processing](#) upload conditions have been met.

DDL handling

When a DDL change is captured, Replicate will close the data file and also create a DFM file if the **Create metadata files in the target folder** option is enabled. When the next batch of changes arrives, Replicate will create a new data file containing the changes. Note that the DFM file created for the new data file will match the new table structure.

Prerequisites

Before you can use Microsoft Azure ADLS as a target endpoint in a Replicate task, the following prerequisites must be met:

- **Permissions:**
- The "Azure Active Directory application ID" specified in the Microsoft Azure ADLS endpoint's **Storage** settings must have write access to the specified ADLS storage target folder.
- In the Access Control (IAM) settings for the ADLS Gen2 file system, assign the "Storage Blob Data Contributor" role to Replicate (AD App ID). It may take a few minutes for the role to take effect.
- **Supported Platforms:** Qlik Replicate on Windows or Linux.



You do not need to install any drivers on the Qlik Replicate Server machine.

Limitations and considerations

The following limitations apply to the Microsoft Azure ADLS target endpoint:

- The following DDLs are supported only: Truncate table, Drop table, Create table, Add Column, Rename Column, Drop Column, and Convert Data Type.
- Full LOB Mode is not supported
- UPDATE and DELETE statements are not supported in Apply Changes replication mode
- Batch optimized apply mode is not supported

- Target lookup is not supported
- The <target folder> parameter cannot include special characters
- Limitations when using Parquet file format:
 - When the [DDL Handling Policy](#) for source tables is set to **Ignore ALTER**, the RENAME COLUMN DDL (on a Primary Key column) is not supported.
 - LOB columns larger than 1 MB are not supported.
- Database names, schema names, or table names containing slash (/) or backslash (\) characters are not supported.
- Proxy:
 - Accessing **Azure Data Lake Storage (ADLS) Gen2** storage via a proxy is not supported with HTTPS protocol.
 - Accessing **Azure Data Lake Storage (ADLS) Gen2** storage via an authenticated proxy is not supported.
- The following Control Tables are not supported as they require UPDATE/DELETE operations, which are not supported:
 - **Replication Status** (requires UPDATE).
Name on target: `attrep_status`
 - **Suspended Tables** (requires DELETE).
Name on target: `attrep_suspended_tables`
For more information on Control Tables, see *Control tables (page 1169)*.

Change data partitioning

When replicating to a Microsoft Azure ADLS target, for each of the source tables, a directory is created under the specified target directory. When *Change Data Partitioning (page 1177)* is enabled, an additional sub-directory is created under the corresponding table directory. The data and metadata (when the metadata option is enabled) files are located in the partition subdirectory, as in the following example:

```
{Target Directory}
  {Table_1}
    {Partition_1}
      Data files
      DFM files
    {Partition_2}
      Data files
      DFM files
    {Partition_3}
      Data files
      DFM files
  {Table_2}
    {Partition_1}
      Data files
      DFM files
    {Partition_2}
      Data files
      DFM files
    {Partition_3}
```

10 Adding and managing target endpoints

Data files
DFM files

Information about the partitions is written to the **attrep_cdc_partitions** Control Table. For information about this table, see *Change data partitions (page 1301)*.

Data types

The following table shows the default mapping from Qlik Replicate data types to Microsoft Azure ADLS target data types. Note that the data type mapping is only relevant if the [Create metadata files in the target folder](#) option is enabled.

For information on source data type mappings, see the section for the source endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

Default data mapping

Qlik Replicate Data Types	Microsoft Azure ADLS Target Data Types
DATE	DATE
TIME	TIME
DATETIME	DATETIME
BYTES	BYTES (length)
BLOB	BLOB
REAL4	REAL4 (7)
REAL8	REAL8 (14)
INT1	INT1 (3)
INT2	INT2 (5)
INT4	INT4 (10)
INT8	INT8 (19)
UINT1	UINT1 (3)
UINT2	UINT2 (5)
UINT4	UINT4 (10)
UINT8	UINT8 (20)
NUMERIC	NUMERIC (p,s)
STRING	STRING (Length)
WSTRING	STRING (Length)
CLOB	CLOB
NCLOB	NCLOB
BOOLEAN	BOOLEAN (1)

Mapping from Qlik Replicate data types to Parquet

When Parquet is set as the message format, due to the limited number of data types supported by Parquet, the data type mappings will be as shown in the table below.

Parquet data type mappings

Replicate Data Type	Parquet Primitive Type	Logical Type
BOOLEAN	BOOLEAN	
INT1	INT32	INT(8, true)
INT2	INT32	INT(16, true)
INT4	INT32	
INT8	INT64	
UINT1	INT32	INT(8, false)
UINT2	INT32	INT(16, false)
UINT4	INT64	
UINT8	INT64	INT(64, false)
REAL4	FLOAT	
REAL8	DOUBLE	
NUMERIC	FIXED_LEN_BYTE_ARRAY (16)	DECIMAL (precision, scale)
STRING	BYTE_ARRAY	STRING
WSTRING	BYTE_ARRAY	STRING
BYTES	BYTE_ARRAY	
BLOB	BYTE_ARRAY	
CLOB	BYTE_ARRAY	STRING
NCLOB	BYTE_ARRAY	STRING
DATE	INT32	DATE
TIME	INT32	TIME(UTC=true, unit=MILLIS)
DATETIME	INT64	TIMESTAMP (UTC=true, unit=MICROS)

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

10 Adding and managing target endpoints

To add an Microsoft Azure ADLS target endpoint to Qlik Replicate:


1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.
2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the endpoint being used.
3. Optionally, in the **Description** field, type a description that helps to identify the endpoint.
4. Select **Target** as the endpoint **role**.
5. Select **Microsoft Azure ADLS** as the endpoint **Type**.
6. Configure the remaining settings in the **General** tab as described in the sections below.




Some of the settings are specific to the selected the storage type (Azure Data Lake Storage (ADLS) Gen2 or Azure Data Lake Storage (ADLS) Gen1).

Azure Storage

ADLS storage

Option	Description
Storage type	Choose one of the following as appropriate: <ul style="list-style-type: none">• Azure Data Lake Storage (ADLS) Gen2• Azure Data Lake Storage (ADLS) Gen1
Storage account	If you selected the Azure Data Lake Storage (ADLS) Gen2 storage type, specify the name of your storage account. <div data-bbox="411 1256 1390 1702"><p><i>To connect to an Azure resource on Government Cloud or China Cloud, you need to specify the full resource name of the storage account. For example, assuming the storage account is "myaccount", then the resource name for China Cloud would be <code>myaccount.dfs.core.chinacloudapi.cn</code></i></p><p><i>In addition, you also need to specify the login URL using the <code>adlsLoginUrl</code> internal parameter. For China Cloud, this would be <code>https://login.chinacloudapi.cn</code></i></p><p><i>For information on setting internal parameters, see Internal Parameters (page 879)</i></p></div>

Option	Description
Data Lake Store name	<p>If you selected the Azure Data Lake Storage (ADLS) Gen1 storage type, specify the full name of the ADLS storage.</p> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> <i>To connect to an Azure resource on Government Cloud or China Cloud, you need to specify the full resource name of the storage account. For example, assuming the storage account is "myaccount", then the resource name for China Cloud would be <code>myaccount.dfs.core.chinacloudapi.cn</code></i></p> <p><i>In addition, you also need to specify the login URL using the <code>adlsLoginUrl</code> internal parameter. For China Cloud, this would be <code>https://login.chinacloudapi.cn</code></i></p> <p><i>For information on setting internal parameters, see Internal Parameters (page 879)</i></p> </div>
Azure Active Directory Tenant ID	The Azure Active Directory tenant ID.
Application Registration Client ID	The application registration client ID.
Application Registration Secret	The application registration secret.
Container	If you selected the Azure Data Lake Storage (ADLS) Gen2 storage type, specify the container containing your folders and files.
SSL CA Path:	If you selected the Azure Data Lake Storage (ADLS) Gen1 storage type, specify the path of a CA certificate (or certificate chain if needed) in PEM format. The CA file must also contain the CA of login.microsoftonline.com.
Target folder	<p>Specify where to create the data files on ADLS.</p> <p>If you selected the Azure Data Lake Storage (ADLS) Gen2 storage type, this should be a location in your file system.</p>




File attributes

Delimiters can be standard characters or a hexadecimal (hex) value. Note that the "0x" prefix must be used to denote a hexadecimal delimiter (e.g. 0x01 = SOH). In the **Field delimiter**, **Record delimiter** and **Null value** fields, the delimiter can consist of concatenated hex values (e.g. 0x0102 = SOHSTX), whereas in the **Quote character** and **Escape character** fields, it can only be a single hex value.



The hexadecimal number 0x00 is not supported (i.e. only 0x01-0xFF are supported).

File attributes

Option	Description
Format	<p>You can choose to create the target files in CSV or JSON, or Parquet format.</p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;">  <i>Parquet format is supported from Replicate May 2022 Service Release 02 only.</i> </div> <p>In a JSON file, each record is represented by a single line, as in the following example:</p> <pre>{ "book_id": 123, "title": "Alice in Wonderland", "price": 6.99, "is_hardcover": false } { "book_id": 456, "title": "Winnie the Pooh", "price": 6.49, "is_hardcover": true } { "book_id": 789, "title": "The Cat in the Hat", "price": 7.23, "is_hardcover": true }</pre> <p>See also: <i>Content-type and content-encoding properties (page 647)</i></p> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;">  <i>Changing the format (for example, from CSV to JSON or from JSON to CSV) while the task is in a stopped state and then resuming the task, is not supported.</i> </div> <div style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;">  <i>If you choose JSON or Parquet format, the following fields will be hidden as they are only relevant to CSV format: Field delimiter, Record delimiter, Null value, Quote character, Escape character, and Add metadata header.</i> </div> <p>For information about data type mappings when using Parquet format and limitations, see <i>Mapping from Qlik Replicate data types to Parquet (page 871)</i> and <i>Limitations and considerations (page 868)</i>.</p>
Field delimiter	<p>The delimiter that will be used to separate fields (columns) in the target files. The default is a comma.</p> <p>Example using a comma as a delimiter:</p> <pre>"mike","male"</pre>
Null value	<p>The string that will be used to indicate a null value in the target files.</p> <p>Example (where \n is the record delimiter and @ is the null value):</p> <pre>"mike","male",295678\n "sara","female",@\n</pre>
Quote escape character	<p>The character used to escape a quote character in the actual data.</p> <p>Example (where" is the quote character and \ is the escape character):</p> <pre>1955,"old, \"rare\", Chevrolet", "\$1000"</pre>

Option	Description
Record delimiter	<p>The delimiter that will be used to separate records (rows) in the target files. The default is a newline (\n).</p> <p>Example:</p> <pre>"mike","male"\n "sara","female"\n</pre>
Quote character	<p>The character that will be used at the beginning and end of a text column. The default is the double-quote character ("). When a column that contains column delimiters is enclosed in double-quotes, the column delimiter characters are interpreted as actual data, and not as column delimiters.</p> <p>Example (where a @ is the quote character):</p> <pre>@mike@,@male@</pre>
Add metadata header	<p>When the target storage format is set to Text, you can optionally add a header row to the data files. The header row can contain the source column names and/or the intermediate (i.e. Replicate) data types.</p> <p>Example of a target file with a header row when both With column names and With data types are selected:</p> <pre>Position:DECIMAL(38,0),Color:VARCHAR(10) 1,"BLUE" 2,"BROWN" 3,"RED" ...</pre>
Maximum file size	<p>The maximum size a file can reach before it is closed (and optionally compressed). This value applies both to data files and to Reference Files.</p> <p>For information on generating reference files, see <i>Setting advanced connection properties (page 876)</i>.</p>
Compress files using	<p>Choose one of the compression options to compress the target files or NONE (the default) to leave them uncompressed. Note that the available compressions options are determined by the selected file format.</p>

Change processing

Change processing

Option	Description
Apply/store changes when: File size reaches	Specify the maximum size of Change Data to accumulate before uploading the file to Microsoft Azure ADLS .
Apply/store changes when: Elapsed time reaches	Specify the maximum time to wait before applying the changes.

Metadata files

Metadata files

Option	Description
Create metadata files in the target folder	<p>When this option is selected, for each data file, a matching metadata file with a .dfm extension will be created under the specified target folder. The metadata files (which are in standard JSON format) provide additional information about the task/data such as the source endpoint type, the source table name, the number of records in the data file, and so on.</p> <p>For a full description of the metadata file as well as possible uses, see <i>Metadata file description (page 1310)</i>.</p>

Testing the connection

To determine if the connection information you entered is correct, click **Test Connection**. If the connection test is successful, click **Save**.



As part of connection testing process, Replicate uploads a test file to the specified [Microsoft Azure ADLS Target folder](#) and then deletes it once a connection has been established.

If the connection is successful, a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

*To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.*

Setting advanced connection properties

In the **Advanced** tab, you can enable the creation of reference files and set post-processing actions. These options are described in detail below.

Post upload processing - Run command after upload

You can process the final target files using a custom command. The command will be run whenever a data file is created.



If the **Generate a reference file** option is selected, a row (specifying the file's location) will be added to the Reference File only after the command completes successfully.

- **Command name** - The location of the command e.g. `C:\utils\move.exe`.
- **Working directory** - The directory where you want the command to run.
- **Parameters** - Specify any parameters that need to be passed to the command during runtime. You can use the following built-in parameters:
 - `${FILENAME}` - The full path to the CSV file containing the full load or CDC data.
 - `${METADATA_FILENAME}` - The full path to the DFM file containing the metadata.For information on creating metadata files, see *Setting general connection properties* (page 871).



- If the CSV/DFM file paths contain spaces, you must enclose these parameters with quotation marks (e.g. `"${FILENAME}"`).
- The post-processing command must return a proper exit code. For information on the available exit code options, see *Exit Codes for the Post-Processing Command* (page 878).

Generate Reference Files - Select this option to generate a Reference File (on Replicate Server) containing the full path to the Apply Changes data files.



The reference file only points to the location of the Apply Changes files, and not the Full Load files.

Reference File(s) folder - The folder on the Replicate machine in which the Reference File will be created.

Example:

`c:\temp\`

Proxy Server

The available settings differ according to the selected storage type (Azure Data Lake Storage (ADLS) Gen1 or Azure Data Lake Storage (ADLS) Gen2).

Settings when using Azure Data Lake Storage (ADLS) Gen1 storage

The following settings are available when using **Azure Data Lake Storage (ADLS) Gen1** storage:

- **Use proxy server:** Select this option to access Microsoft Azure ADLS via a proxy server.
- **Host name:** The host name of the proxy server.
- **Port:** The port via which to access the proxy server.
- **User name:** The user name for accessing the proxy server.
- **Password:** The password for accessing the proxy server.
- **Scheme:** Select which protocol to use to access the server (HTTP or HTTPS).



The selected Scheme applies to the storage, but does not apply to the ODBC connection.

- **SSL CA Path** - The location of the CA file on the Replicate Server machine when **HTTPS** is the selected **Scheme**.

Settings when using Azure Data Lake Storage (ADLS) Gen2 storage

The following settings are available when using **Azure Data Lake Storage (ADLS) Gen2** storage:



Proxy authentication and the HTTPS scheme are not supported when using Azure Data Lake Storage (ADLS) Gen2 storage.

- Connect via proxy server to:
 - **Cloud storage:** Select to connect to Azure Data Lake Storage (ADLS) Gen2 storage via a proxy server.
 - **Azure Active Directory:** Select to connect to Azure Active Directory via a proxy server.



*When **Azure Active Directory** is selected, the connection is established via <https://login.microsoftonline.com/>.*

- **Host name:** The host name of the proxy server.
- **Port:** The port via which to access the proxy server.

Exit Codes for the Post-Processing Command

The post-processing command must return a proper exit code. You can either use the standard exit code values described below or set a custom exit code value as described in **Setting Post Command exit codes with an internal parameter** below.

Standard Post Command exit codes

- **0** - Success
- **1** - Recoverable error. The task will recover from the point of failure according to the settings in the *Environmental errors (page 1187)* tab.
- **2** - Table error. If a table error occurs, Replicate will handle the error according to the settings in the *Table Errors (page 1191)* tab.
- **3** (or any other value e.g. -100) - Fatal error. The task will fail and not attempt recovery.

Setting Post Command Exit Codes with an Internal Parameter

You can use internal parameters to set exit codes with custom values. This is especially useful if your application already uses the standard exit code values.

See **Standard Post Command exit codes** above for a description of the exit codes.

- `successExitCode`
- `recoverableErrorExitCode`

- `tableErrorExitCode`
- `fatalErrorExitCode`

Internal Parameters

Internal parameters are parameters that are not exposed in the UI and should only be used if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings Summary

You can view a summary of your setting by clicking the View Setting Summary link. This is useful if you need to send a summary of your setting to Qlik Support.

Generating reference files

In the **Advanced** tab of the Microsoft Azure ADLS target endpoint, you can enable the **Generate a reference file** option. The Reference File contains a list of the Change File locations and is therefore only relevant if the task's Apply Changes or Store Changes options are enabled.

The format of the reference file name is as follows:

```
<microsoft_azure_adls_target_endpoint_display_name><counter>.csv|json
```

Example:

```
MyAzureADLS00000001.csv
```



The counter suffix increases incrementally each time a new Reference File is generated (i.e. which occurs when the file reaches the maximum size defined [in the General tab](#)). Once a new Reference File has been generated, you can delete the old reference file(s) if required.

10 Adding and managing target endpoints

Whenever an Apply Changes data file is created, a new row is added to the Reference File in the following format:

<Source_Table_Name>, <Data_Lake_Store_name>/<path>/<file_name>

Example:

employees,mydatalakestore/new/files/my.company/20170611-120144192.csv

Note that if the **Run command after upload** option in the **Advanced** tab is also enabled, the Reference File will be generated after the post-processing completes.

10.28 Using Microsoft Azure Database for MySQL as a target

This section describes how to set up and use a Microsoft Azure Database for MySQL target endpoint in a replication task.

In this section:

- *Prerequisites (page 882)*
- *Limitations and considerations (page 883)*
- *Security requirements (page 883)*
- *Supported data types (page 884)*
- *Setting general connection properties (page 886)*
- *Setting advanced connection properties (page 887)*

Prerequisites

Before using Microsoft Azure Database for MySQL as an endpoint in a Qlik Replicate task, make sure that the prerequisites outlined below have been met.

Driver setup

Installing the driver on Windows

Install MySQL ODBC Unicode Driver 64-bit version 8.0.32 on the Replicate machine.

Installing the driver on Linux

1. Stop the Replicate service:

```
./areplicate stop
```
2. Optionally, confirm that the service has stopped:

```
./areplicate status
```
3. Install MySQL ODBC Unicode Driver 64-bit version 8.0.32 on the Replicate machine.
4. Change the working directory to `<Replicate-Install-Dir>/bin`.
5. Copy the driver location to the `site_arep_login.sh` file:

```
echo "export LD_LIBRARY_PATH=\$LD_LIBRARY_PATH:/usr/lib64" >> site_arep_login.sh
```

This will add the driver to "LD_LIBRARY_PATH" and update the driver location in the **site_arep_login.sh** file.
6. Optionally, confirm that the driver location was copied:

```
cat site_arep_login.sh
```
7. Make sure that the `/etc/odbcinst.ini` file contains an entry for MySQL, as in the following example:

```
[MySQL ODBC 8.0 Unicode Driver]  
Driver = /usr/lib64/libmyodbc8w.so  
UsageCount = 1
```
8. Start the Replicate service:

```
./areplicate start
```
9. Optionally confirm that the service has started:

```
./areplicate status
```

General prerequisites

Make sure that the following prerequisites have been met:

- A Microsoft Azure Database for MySQL account with the required *Security requirements (page 883)*.
- A Microsoft Azure Database for MySQL database with the tables that you want to replicate should be accessible in your network.

The following Microsoft Azure Database for MySQL editions are supported:

- Microsoft Azure Database for MySQL Community Edition
- Microsoft Azure Database for MySQL Standard Edition
- Microsoft Azure Database for MySQL Enterprise Edition
- Microsoft Azure Database for MySQL Cluster Carrier Grade Edition
- In order to work with Microsoft Azure Database for MySQL 8 as a target endpoint, you need to:
 - Set the parameter `secure_file_priv = NULL` in the Microsoft Azure Database for MySQL database.

- If you are using LOB columns, set the parameter `sort_buffer_size = 128M` in the Microsoft Azure Database for MySQL database.

Replicating 4-byte UTF8 emojis

To support replication of 4-byte UTF8 emojis, the target schema character set must be set to `utf8mb4`.

Limitations and considerations

The following limitations apply:

- Due to the way Microsoft Azure Database for MySQL operates, when loading data to a Microsoft Azure Database for MySQL target during a Full Load task, duplicate key errors will not be reported to the logs.
- When updating a column's value to its existing value, a zero rows affected is returned from Microsoft Azure Database for MySQL (unlike Oracle and Microsoft SQL Server that perform an update of one row). This generates an entry in the `attrep_apply_exceptions` control table and the following warning: `some changes from the source database had no impact when applied to the target database. see attrep_apply_exceptions table for details.`
- Due to an issue with MySQL 5.7 (which was resolved in version 8.0), Replicate cannot insert multibyte characters into TEXT columns when the Full Load uses CSV files. Consequently, if the target endpoint is MySQL 5.7 and a replicated table contains TEXT columns with UTF-8 multibyte characters, the target table might be empty.

Workaround:

In the **Advanced** tab, set the `loadUsingCSV` internal parameter to `FALSE`. Note that this might impact performance.

- If you need to replicate 4-byte emoji characters, the target schema character set must be set to `utf8mb4`.

Security requirements

You must provide Microsoft Azure Database for MySQL account access to the Qlik Replicate user. This user must have read/write privileges in the Microsoft Azure Database for MySQL database.

Additionally, for each target schema involved in the replication, the Qlik Replicate user must have the following permissions:

- ALTER
- CREATE
- CREATE TEMPORARY TABLES
- DELETE
- DROP
- INSERT
- SELECT
- UPDATE

Supported data types

The following table shows the Microsoft Azure Database for MySQL database target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

Supported data types

Qlik Replicate Data Types	Microsoft Azure Database for MySQL Data Types
BOOL	BOOL
BYTES	If length is => 1 and =< 8095, then: VARBINARY (Length) If length is => 8096 and =< 65535, then: BLOB If length is => 65536 and =< 16777215, then: MEDIUMBLOB If length is => 16777216 and =< 2147483647, then: LONGBLOB
DATE	DATE
TIME	TIME
DATETIME	If scale is => 0 and =< 6, then: DECIMAL (p,s) If scale is => 7 and =< 12, then: VARCHAR (37)
INT1	TINYINT
INT2	SMALLINT
INT4	INTEGER
INT8	BIGINT

10 Adding and managing target endpoints

Qlik Replicate Data Types	Microsoft Azure Database for MySQL Data Types
NUMERIC	<p>If scale is \Rightarrow 0 and \leq 30, then:</p> <p>DECIMAL (p,s)</p> <p>If scale is \Rightarrow 31 and \leq 100, then:</p> <p>VARCHAR (45)</p>
REAL4	FLOAT
REAL8	DOUBLE
STRING	<p>If length is \Rightarrow 1 and \leq 8095, then:</p> <p>VARCHAR (Length)</p> <p>If length is \Rightarrow 8096 and \leq 65535, then:</p> <p>TEXT</p> <p>If length is \Rightarrow 65536 and \leq 16777215, then:</p> <p>MEDIUMTEXT</p> <p>If length is \Rightarrow 16777216 and \leq 2147483647, then:</p> <p>LONGTEXT</p>
UINT1	UNSIGNED TINYINT
UINT2	UNSIGNED SMALLINT
UINT4	UNSIGNED INTEGER
UINT8	UNSIGNED BIGINT
WSTRING	<p>If length is \Rightarrow 1 and \leq 8095, then:</p> <p>VARCHAR (Length)</p> <p>If length is \Rightarrow 8096 and \leq 65535, then:</p> <p>TEXT</p> <p>If length is \Rightarrow 65536 and \leq 16777215, then:</p> <p>MEDIUMTEXT</p> <p>If length is \Rightarrow 16777216 and \leq 2147483647, then:</p> <p>LONGTEXT</p>

Qlik Replicate Data Types	Microsoft Azure Database for MySQL Data Types
BLOB	<p>When the Allow unlimited LOB size option is enabled:</p> <ul style="list-style-type: none"> • LONGBLOB <p>When the Limit LOB size to option is enabled:</p> <ul style="list-style-type: none"> • If the specified value is 63 KB or smaller, then: BLOB • If the specified value is 64 KB or larger, then: LONGBLOB <p>For information on limiting LOB size, see <i>Target metadata (page 1166)</i>.</p>
NCLOB	<p>When the Allow unlimited LOB size option is enabled:</p> <ul style="list-style-type: none"> • TEXT <p>When the Limit LOB size to option is enabled:</p> <ul style="list-style-type: none"> • If the specified value is 63 KB or smaller, then: TEXT • If the specified value is 64 KB or larger, then: LONGTEXT <p>For information on limiting LOB size, see <i>Target metadata (page 1166)</i>.</p>
CLOB	<p>When the Allow unlimited LOB size option is enabled:</p> <ul style="list-style-type: none"> • TEXT CHARACTER SET UTF-16 <p>When the Limit LOB size to option is enabled:</p> <ul style="list-style-type: none"> • If the specified value is 63 KB or smaller, then: TEXT CHARACTER SET UTF-16 • If the specified value is 64 KB or larger, then: LONGTEXT CHARACTER SET UTF-16 <p>For information on limiting LOB size, see <i>Target metadata (page 1166)</i>.</p>

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a Microsoft Azure Database for MySQL target endpoint to Qlik Replicate:

1. In the Qlik Replicate Console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.

- In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
- In the **Description** field, type a description that helps to identify the Microsoft Azure Database for MySQL database. This is optional.
- Select **TARGET** as the database **role**.
- From the **Type** drop-down list, select **Microsoft Azure Database for MySQL**.
- In the **Server** field, enter the host name or IP address of the computer on which the Microsoft Azure Database for MySQL database is installed.



Notes

- This information is case sensitive.*
- To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**. If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box. To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.*

- Optionally, change the default port (3306).
- Type the Microsoft Azure Database for MySQL authentication information (**User Name**, **Password**) for the authorized user for this database. If you do not know this information, see your database Administrator (DBA).
- Select one of the following **Load source schemas into** options:
 - The following database** - When this option is selected, all source schemas will be loaded into the selected database.
 - Multiple endpoints** - When this option is selected, each of the source schemas will be loaded into its corresponding database.

Setting advanced connection properties

In the **Advanced** tab, you can set the following properties:

- Max file size (KB):** Select or type the maximum size (in KB) of a CSV file before it is loaded into the target database. The default value is 32000 KB.
- Use parallel loading:** Select this option to improve performance when loading data into the target database.
 - Use the following number of threads:** Specify how many threads to use to load the data into the Microsoft Azure Database for MySQL target database. Note that setting a large number of threads may have an adverse effect on database performance since a separate connection is required for each thread.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.29 Using Microsoft Azure Database for PostgreSQL as a target

This section describes how to set up and use a Microsoft Azure Database for PostgreSQL target endpoint in a replication task.


In this section:

- *Prerequisites (page 890)*
- *Limitations (page 992)*
- *Security requirements (page 891)*
- *Microsoft Azure Database for PostgreSQL target data types (page 891)*
- *Setting general connection properties (page 893)*
- *Setting advanced connection properties (page 894)*

Prerequisites

This topic describes the client prerequisites when replicating to a Microsoft Azure Database for PostgreSQL target.

Qlik Replicate Server for Windows

- **The PostgreSQL ODBC Driver:** PostgreSQL ODBC Driver [psqlodbc_13_01_0000-x64](#) or above must be installed on the Qlik Replicate machine.
- **PgAdmin:**
 1. Download and install  [PgAdmin 4 \(Windows\)](#) on the Qlik Replicate machine.
 2. Add the "psql.exe" path to the system PATH.
- Microsoft Visual Studio 2013 Redistributable (x86) - *not* (x64) - must be installed on the Qlik Replicate machine.

Qlik Replicate Server for Linux

Download PostgreSQL 13 ODBC driver, and client programs and libraries.

- **Example PostgreSQL ODBC file name:** postgresql13-odbc-13.02.0000-1PGDG.rhel8.x86_64.rpm
- **PostgreSQL client programs and libraries file name:** postgresql13-13.8-1PGDG.rhel<version>.x86_64.rpm

Then, open a shell prompt and do the following:

1. Stop the Replicate service:

```
./areplicate stop
```
2. Optionally, confirm that the service has stopped:

```
./areplicate status
```
3. Install PostgreSQL 13 client programs and libraries.
4. Install the ODBC driver for PostgreSQL 13.
5. Change the working directory to `<Replicate-Install-Dir>/bin`.
6. Copy the driver location to the `site_arep_login.sh` file:

```
echo "export LD_LIBRARY_PATH=\$LD_LIBRARY_PATH:/usr/lib:/usr/lib64:/usr/pgsql-13/lib" >> site_arep_login.sh
```

This will add the driver to "LD_LIBRARY_PATH" and update the driver location in the **site_arep_login.sh** file.
7. Optionally, confirm that the driver location was copied:

```
cat site_arep_login.sh
```
8. Makes sure that the `/etc/odbcinst.ini` file contains an entry for PostgreSQL, as in the following example:

```
[PostgreSQL Unicode(x64)]
Description = PostgreSQL ODBC driver
Driver = /usr/pgsql-13/lib/psqlodbc.so
Setup = /usr/pgsql-13/lib/psqlodbcw.so
UsageCount = 1
```
9. Start the Replicate service:

```
./areplicate start
```

10. Optionally, confirm that the service has started:

```
./areplicate status
```

Limitations

- UPDATE and DELETE operations on tables without a Primary Key/Unique Index that contain duplicate records is not supported. Any changes to these tables will affect all the records in the target.
- Tables with a LOB column in the key are not supported in Batch Optimized Apply mode. When a VARBINARY column is used as a source table key, a BYTEA column will be created in the target. This causes unpredictable behavior in Batch Optimized Apply mode. As a workaround, we suggest using the SQLite HEX function to convert VARBINARY to VARCHAR.

Security requirements

The user specified in the **General** tab when *Setting general connection properties (page 893)* must be a registered user in the Microsoft Azure Database for PostgreSQL database.

In addition, the following privileges must be granted:

- Can login
- Create databases



If the target schema already exists in the database, the "Create databases" privilege is not required.

Microsoft Azure Database for PostgreSQL target data types

The Microsoft Azure Database for PostgreSQL endpoint for Qlik Replicate supports most Microsoft Azure Database for PostgreSQL database data types. The following table shows the Microsoft Azure Database for PostgreSQL database target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types. Unsupported data types are listed below the table.

For information on how to view the data type that is mapped from the source, see the section for the source database you are using. For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

Supported data types

Qlik Replicate Data Types	Microsoft Azure Database for PostgreSQL Data Types
BOOL	BOOL
BYTES	BYTEA
DATE	DATE
TIME	TIME

Qlik Replicate Data Types	Microsoft Azure Database for PostgreSQL Data Types
DATETIME	<p>If scale is => 0 and =< 6, then:</p> <p>TIMESTAMP</p> <p>If scale is => 7 and =< 12, then:</p> <p>VARCHAR (37)</p>
INT1	SMALLINT
INT2	SMALLINT
INT4	INTEGER
INT8	BIGINT
NUMERIC	DECIMAL (P, S)
REAL4	FLOAT4
REAL8	FLOAT8
STRING	<p>If length is 1 - 21845, then:</p> <p>VARCHAR (Length in Bytes = The STRING value multiplied by three)</p> <p>If length is 21846 - 2147483647, then:</p> <p>VARCHAR (65535)</p>
UINT1	SMALLINT
UINT2	INTEGER
UINT4	BIGINT
UINT8	BIGINT
WSTRING	<p>If length is 1 - 21845, then:</p> <p>VARCHAR (Length in Bytes = The WSTRING value multiplied by three)</p> <p>If length is 21846 - 2147483647, then:</p> <p>VARCHAR (65535)</p>
BLOB	BYTEA
NCLOB	TEXT
CLOB	TEXT

Data types when replicating from a PostgreSQL source

When replicating from a PostgreSQL source, the target table will be created with the same data types for all columns, apart from columns with user-defined or PostGIS data types. In such cases, the data type will be created as "character varying" in the target.

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a Microsoft Azure Database for PostgreSQL target endpoint to Qlik Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints* (page 114).
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Microsoft Azure Database for PostgreSQL database. This is optional.
4. Select **Target** as the database **role**.
5. From the **Type** drop-down list, select **Microsoft Azure Database for PostgreSQL**.
6. Type the **Server** name. This is the name or IP address or host name of the computer with the Microsoft Azure Database for PostgreSQL database that you want to access.
7. Optionally, change the default port (5432).
8. Enter the Microsoft Azure Database for PostgreSQL database authentication information (**User name**, **Password**) of an authorized Microsoft Azure Database for PostgreSQL user. If you do not know this information, see your database system manager.



Consider the following:

- *This information is case sensitive.*
- *To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**. If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box. To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.*



Make sure that the specified Microsoft Azure Database for PostgreSQL database user has the correct access privileges.

9. Type the **Database name** or select one from the list of available endpoints. This is the name of the Microsoft Azure Database for PostgreSQL database to which you are replicating data.

SSL authentication options

In the **Security** section, you can configure Replicate to connect to the Microsoft Azure Database for PostgreSQL database using SSL.



The specified certificates are relevant only for bulk insert operations.

- **SSL Mode:** Select one of the following:
 - **Disable** - Connect with a surname and password only.
 - **Allow** - Establish an encrypted connection if requested by the server.
 - **Prefer** - Establishes an encrypted connection if the server supports encrypted connections, falling back to an unencrypted connection if an encrypted connection cannot be established.
 - **Require** - Establishes an encrypted connection if the server supports encrypted connections. The connection attempt fails if an encrypted connection cannot be established.
 - **Verify CA** - Similar to **Required**, but also verifies the server Certificate Authority (CA) certificate against the configured CA certificates. The connection attempt fails if no valid matching CA certificates are found.
 - **Verify Full** - Similar to **Verify CA**, but also performs host name identity verification by checking the host name the client (i.e. Replicate) uses for connecting to the server against the identity in the certificate that the server sends to the client. The client checks whether the host name that it uses for connecting matches the Common Name value in the server certificate. The connection fails if there is a mismatch.
- **Client certificate path** - The path to the client certificate requested by the server.
- **Secret key path** - The path to the client private key file in PEM format.
- **CA path** - The path to the Certificate Authority (CA) that issued the client certificate file in PEM format.
- **CRL path** - The path to the CRL certificate. This file contains certificates revoked by certificate authorities. If the server certificate appears in this list, the connection will fail.
- **SSL compression** - Select this option to compress the data before it is encrypted.

Setting advanced connection properties

In the **Advanced** tab, you can set the following properties:

You can set the following properties:

- **Max file size (KB):** Select or type the maximum size (in KB) of a CSV file before the file is loaded into the Microsoft Azure Database for PostgreSQL target database. The default value is 32000 KB.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.

4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.30 Using Microsoft Azure Event Hubs as a target

This section describes how to set up and use Microsoft Azure Event Hubs as a target endpoint in a replication task. In a task with a Microsoft Azure Event Hubs target endpoint, each source record is transformed into a message which is then written (with an optional message key) to a partition in the specified hub.

In this section:

- *Prerequisites (page 897)*
- *Transaction processing by the consumer (page 897)*
- *Limitations (page 898)*
- *Supported target data types (page 899)*
- *Setting general connection properties (page 901)*
- *Setting advanced connection properties (page 904)*
- *The Replicate envelope (page 905)*
- *Metadata and data messages (page 907)*

Prerequisites

Before you can use Microsoft Azure Event Hubs as a target endpoint in a Replicate task, the following prerequisites must be met:

- The target hubs must already exist before starting the replication task. Note that if you intend to use the [Separate hub for each table](#) option in the endpoint settings, the hub name must adhere to the following format:
`SourceSchemaName . HubName`
where `HubName` must be identical to the source table name.
Example:
`HR . Employees`
- Create a hub named **attrep_apply_exceptions** before starting the replication task. Note that if this hub does not exist, the task will always fail when it encounters a data error, regardless of the [error handling policy](#).
For a description of the **attrep_apply_exceptions** table, see *Apply exceptions (page 1298)*
- To be able to browse for hubs (in the [General tab](#)), the namespace Shared Access Policy must have "Manage" permission.
- If the namespace shared access policy does not have "Manage" permission, then the hub Shared Access Policy must have at least "Send" permission.

Transaction processing by the consumer

When configuring the Qlik Replicate Microsoft Azure Event Hubs endpoint, users can configure various settings that affect where messages are published within the Microsoft Azure Event Hubs infrastructures (hubs/partitions).

During a task's CDC stage, committed changes that are detected by the Qlik Replicate source endpoint are grouped by transaction, sorted internally in chronological order, and then propagated to the target endpoint. The target endpoint can handle the changes in various ways such as applying them to the target tables or storing them in dedicated Change Tables.

Each CDC message has both a transaction ID as well as change sequence. As the change sequence is a monotonically growing number, sorting events by change sequence always achieves chronological order. Grouping the sorted events by transaction ID then results in transactions containing chronologically sorted changes.

However, as Microsoft Azure Event Hubs is a messaging infrastructure, applying changes is not feasible while storing changes in tables is meaningless. The Replicate Microsoft Azure Event Hubs endpoint, therefore, takes a different approach, which is to report all transactional events as messages.

How it works

Each change in the source system is translated to a data message containing the details of the change including the transaction ID and change sequence in the source. The data message also includes the changed columns before and after the change. As explained above, the order in which the Microsoft Azure Event Hubs target writes the messages is the same as order of changes within each transaction.

Once a data message is ready to be sent to Microsoft Azure Event Hubs, the hub and partition it should go to are determined by analyzing the [endpoint settings](#) as well as potentially [transformation settings](#). For example, the user might decide to configure the endpoint in such a way that every table is sent to a different hub and set the partition strategy to "Random", meaning that each message (within the same table) will be sent to a different partition.

Transaction consistency from a consumer perspective

If maintaining transaction consistency is important for the consumer implementation, it means that although the transaction ID exists in all data messages, the challenge is to gather the messages in a way that would facilitate identifying a whole transaction. An additional challenge is getting the transaction in the original order they were committed, which could be an even greater challenge if transactions are spread across multiple hubs and partitions.

The simplest way of achieving the above goal is to direct Replicate to a specific hub and a specific partition (in the [endpoint settings](#)). This means that all data messages will end up in a single partition, thus guaranteeing ordered delivery both of transactions *and* of changes within a transaction. The consuming application could then consume messages - accumulating a transaction in some intermediate memory buffer - and when a new transaction ID is detected, mark the previous transaction as completed.

Although the simple way may work, it's not very efficient at the task level as all messages end up in the same hub and partition, not necessarily utilizing the full parallelism of the Microsoft Azure Event Hubs cluster. This may be a non-issue if there are multiple tasks, each taking advantage of a different hub/partition. In such as scenario, the gathering of messages from those tasks may very well utilize the cluster optimally.

The more generic way where data may be spread over multiple hubs and partitions means that some intermediate buffer such as memory, a table in a relational database, or even other hubs would need to be used to collect information about transactions. Then, the transactions would need to be rebuilt by periodically (every few minutes/hours) sorting the events collected from Replicate's Microsoft Azure Event Hubs output by the change sequence and grouping them by transaction ID.

Limitations

When defining a task with Microsoft Azure Event Hubs as the target endpoint, the following limitations apply:

- The Microsoft Azure Event Hubs target endpoint does not support unlimited LOB size. Therefore, when replicating from source tables with LOB columns, *do not* select the **Allow unlimited LOB size** option. For more information on defining LOB settings, see *Target metadata (page 1166)*.
- **Batch optimized apply** mode is not supported. If this mode is set, the task will automatically switch to **Transactional apply** mode and issue an appropriate warning. For more information on these modes, see *Change Processing Tuning (page 1181)*.
- Store Changes replication mode is not supported. For more information on Store Changes mode, see *Adding tasks (page 110)*.
- The [Ignore ALTER Apply Changes](#) setting is not supported for changes to source data types and table renaming.
- Column names must begin with [A-Za-z_] (letters or an underscore) followed by [A-Za-z0-9_] (letters, digits, or an underscore). For example, **_Test_** is a valid column name whereas **&Test** is not.

10 Adding and managing target endpoints

If a source column name does not adhere to this rule, then [a transformation should be used](#) to rename the column.

- The [Drop and Create table Target Table Preparation](#) option is not supported
- The [Truncate before loading Target Table Preparation](#) option is not supported
- The DDL History and Change Data Partitioning [Control tables](#) are not supported
- DDL Changes: Dropping or renaming a table is not supported

Supported target data types

The following table shows the default mapping from Qlik Replicate data types to Microsoft Azure Event Hubs data types.

For information on source data type mappings, see the section for the source endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.





When using the JSON message format, binary values are represented as hexadecimal digits.

Qlik Replicate and Microsoft Azure Event Hubs data types

Qlik Replicate Data Types	Microsoft Azure Event Hubs Target Data Types in Schema Messages
DATE	DATE
TIME	TIME
DATETIME	DATETIME
BYTES	BYTES (length)
BLOB	BLOB
REAL4	REAL4 (7)
REAL8	REAL8 (14)
INT1	INT1 (3)
INT2	INT2 (5)
INT4	INT4 (10)
INT8	INT8 (19)
UINT1	UINT1 (3)
UINT2	UINT2 (5)

10 Adding and managing target endpoints

Qlik Replicate Data Types	Microsoft Azure Event Hubs Target Data Types in Schema Messages
UINT4  Values larger than $2^{31}-1$ are not supported.	UINT4 (10)
UINT8  Values larger than $2^{63}-1$ are not supported.	UINT8 (20)
NUMERIC	NUMERIC (p,s)
STRING	STRING (Length)
WSTRING	STRING (Length)
CLOB	CLOB
NCLOB	NCLOB
BOOLEAN	BOOLEAN (1)

Mapping to JSON and Avro

Avro message format uses logical types for a more precise representation of the data type.

Data type mappings

Qlik Replicate Data Types	JSON or Avro Primitive Data Types	Avro Logical Data Types
DATE	STRING	DATE Annotates an Avro INT.
TIME	STRING	TIME-MILLIS Annotates an Avro INT.
TIMESTAMP	STRING	TIMESTAMP-MICROS Annotates an Avro LONG.
STRING	STRING	-
WSTRING	STRING	-
CLOB	STRING	-
NCLOB	STRING	-

Qlik Replicate Data Types	JSON or Avro Primitive Data Types	Avro Logical Data Types
NUMERIC	STRING	DECIMAL (p,s) Annotates an Avro BYTES.
BYTES	BYTES	-
BLOB	BYTES	-
REAL4	FLOAT	-
REAL8	DOUBLE	-
INT1	INT	-
INT2	INT	-
INT4	INT	-
UINT1	INT	-
UINT2	INT	-
UINT4	LONG	-
INT8	LONG	-
UINT8	STRING	DECIMAL (20,0) Annotates an Avro BYTES.
BOOLEAN	BOOLEAN	-

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To define the general connection properties:

1. Click the **Manage Endpoint Connections** toolbar button.
The **Manage Endpoints Connections** dialog box opens.
2. Click the **New Endpoint Connection** toolbar button.
The **Name**, **Description**, **Type** and **Role** fields are displayed on the right.
3. In the **Name** field, specify a display name for the endpoint.
4. In the **Description** field, optionally type a description for the Microsoft Azure Event Hubs endpoint.
5. Select **Target** as the endpoint **Role**.
6. Select **Microsoft Azure Event Hubs** as the endpoint **Type**.
The dialog box is divided into **General** and **Advanced** tabs.
7. In the **Access Details** section, set the following properties:
 - **Namespace:** Enter the name of your Event Hubs namespace.
Example: `eventhubdemo`



To connect to an Azure resource on Government Cloud or China Cloud, you need to specify the full resource name of the namespace, which is the name plus the suffix. For instance, assuming the namespace name is `eventhubdemo`, then the full resource name for a public resource would be `eventhubdemo.servicebus.windows.net`. For Government Cloud or China Cloud, the format should be the same but with the Government Cloud or China Cloud suffix instead.

For information on setting internal parameters, see *Setting advanced connection properties* (page 904)

- **Shared Access Policy Level:** Select either **Namespace level** or **Event Hub level** according to the level defined for your Shared Access Policy.
- **Shared Policy Name:** Enter the name of your shared access policy.
Example: `RootManageSharedAccessKey`
- **Shared Access Key:** Enter your shared access primary or secondary key.
Example: `BZLreXGxiWiRpGAog9Zf6b3K7ycRsImfBWqsR+SJp34=`

8. In the **Message Properties** section, select either **JSON** or **Avro** as the message format.



Qlik provides an Avro Message Decoder SDK for consuming Avro messages produced by Qlik Replicate. You can download the SDK as follows:

1. Go to [Product Downloads](#).
2. Select **Qlik Data Integration**.
3. Scroll down the **Product** list and select **Replicate**.
4. In the **Download Link** column, locate the **QlikReplicate_<version>_Avro_Decoder_SDK.zip** file. Before starting the download, check the **Version** column to make sure that the version correlates with the Replicate version you have installed.
5. Proceed to download the **QlikReplicate_<version>_Avro_Decoder_SDK.zip** file.

For usage instructions, see *Kafka Avro consumers API* (page 1339).

An understanding of the Qlik envelope schema is a prerequisite for consuming Avro messages produced by Qlik Replicate. If you do not wish to use the SDK, see [The Qlik Envelope](#) for a description of the Qlik envelope schema.

9. Optionally, in the **Namespace Settings** section, adjust the number of **Throughput units**. Increasing the number of throughput units may improve performance in certain scenarios.
10. In the **Data Message Publishing** section, set the following properties:
 - a. In the **Publish the data to** field, choose one of the following:
 - **Specific hub** - to publish the data to a single hub. Either type a hub name or use the browse button to select the desired hub.
 - **Separate hub for each table** - to publish the data to multiple hubs corresponding to the source table names. If you select this option, the hub name format must be [as described in the prerequisites](#).



Microsoft Azure Event Hubs supports a maximum of ten hubs, which includes hubs that are required for storing Replicate [Control Table](#) data. At least one of the ten hubs must be set aside for the mandatory `attrep_apply_exceptions` Control Table, leaving nine hubs to which source data can be published (providing that no other Control Tables are set).

The target hub name consists of the source schema name and the source table name, separated by a period (e.g. "dbo.Employees"). The format of the target hub name is important as you will need to prepare these hubs in advance.

- b. From the **Partition strategy** drop-down list, field, select either **Random** or **By message key**. If you select **Random**, each message will be written to a randomly selected partition. If you select **By message key**, messages will be written to partitions based on the selected **Message key** (described below).
- c. From the **Message key** drop-down list, field, select one of the following:



Note The message key is represented as a string, regardless of the selected data message format (JSON/Avro).

- **Schema and table name** - For each message, the message key will contain a combination of schema and table name (e.g. "dbo+Employees").
When **By message key** is selected as the **Partition strategy**, messages consisting of the same schema and table name will be written to the same partition.
 - **Primary key columns** - For each message, the message key will contain the value of the primary key column.
When **By message key** is selected as the **Partition strategy**, messages consisting of the same primary key value will be written to the same partition.
11. In the **Metadata Message Publishing** section, specify whether or where to publish the message metadata.

From the **Publish** drop-down list, select one of the following options:

- **Do not publish metadata messages** - When this option is selected, only data messages will be published.
- **Publish metadata messages to a dedicated metadata hub**
If you select this option, either type the **Hub name** or use the Browse button to select the desired hub. Note that the Browse button will only be available if the **Shared Access Policy Level** described above is set to **Namespace level**.



It is strongly recommended not to publish metadata messages to the same stream as data messages.

Overriding the default settings

A transformation can be defined that overrides the topic, partition and message key settings defined in the **General** tab.



Before you can define such a transformation, you first need to add a source endpoint to the task and select the tables you want to replicate.

To define a transformation:

1. Open the task you defined.
2. If you are defining a transformation for a single table, select one of the source tables. Otherwise, skip to Step 3.
3. Define a transformation that adds one of the following columns:



The columns listed below (prefixed with a *\$*) instruct Replicate to route the message to the desired hub and/or partition, and will not be included in the actual message itself.

- *\$hub* - To write messages to a specific hub.
- *\$partition* - To write messages to a specific partition.
- *\$key* - To create a custom message key.

For information on creating a transformation for a single table, see *Defining transformations for a single table/view (page 1084)*.

For information on creating a global transformation rule, see *Starting the Global Transformation Rules wizard (page 1107)*.

4. Define an expression for the new column that returns the following values:
 - For a *\$hub* column, the expression should return the topic name.
 - For a *\$partition* column, the expression should return the partition number. Note that an error will be returned during runtime if the partition number does not exist.
 - For a *\$key* column, the expression should return the message key contents. Note that the expression must return a non-empty value.

For information on creating expressions, see *Using the Expression Builder (page 1137)*.

Setting advanced connection properties

In the **Advanced** tab, you can set the following advanced properties for the Microsoft Azure Event Hubs target endpoint:

Connection properties

Option	Description
Message Maximum Size	In the Message maximum size field, specify the maximum size of messages that the namespace is configured to receive (<code>message.max.bytes</code>). Replicate will not send messages larger than the maximum size.
Use proxy server	Select this option to access the namespace via a proxy server. <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> Replicate uses AMQP protocol to access the namespace. </div>

Option	Description
Host name	The host name of the proxy server.
Port	The port via which to access the proxy server.
User name	The user name for accessing the proxy server.
Password	The password for accessing the proxy server.
Scheme	Select which protocol to use to access the server (HTTP or HTTPS). In order to use HTTPS, you must first install the CA certificate that signed the proxy's certificate on the Replicate Server machine. Do this as follows: <ul style="list-style-type: none">• On Windows: Add the CA certificate to the Trusted Root Certification Authorities store of Local Computer• On Linux: Add the CA certificate to <code>/etc/pki/tls/certs/ca-bundle.crt</code>

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

The Replicate envelope

All Replicate message types covered in this section are encapsulated in a single message schema called the Replicate envelope. The schema of the Replicate envelope is as following:

```
{  
  "type": "record",  
  "name": "MessageEnvelope",  
}
```

```
"fields": [
  {"name": "magic", "type": {"type": "fixed", "name": "Magic", "size": 5}},
  {"name": "type", "type": "string"},
  {"name": "headers", "type": ["null", {"type": "map", "values": "string"}]},
  {"name": "messageSchemaId", "type": ["null", "string"]},
  {"name": "messageSchema", "type": ["null", "string"]},
  {"name": "message", "type": "bytes"}
]
```

The fields in the envelope are as follows:

- **magic** (5 bytes fixed field)
The constant "atMSG" is used to identify this form of message. The "atMSG" constant should be used to validate that this message is indeed an Replicate envelope message.
- **type** (string field)
Describes the enveloped message type. This can be one of two values: MD which stands for metadata message and DT which stands for data message.
- **headers** (map of string key and value)
A free for use map for various properties set at the application level. Currently, no headers are set by Qlik Replicate but this may change in future versions.
- **messageSchemaId** (null or string)
A reference to a schema defined elsewhere, which can be used to deserialize the bytes in the message field. This specification does not explain how the schema ID is used for looking up the actual schema - it is an application level detail. This field is used exclusively with the messageSchema field.
- **messageSchema** (null or string)
An embedded UTF-8 encoded Avro JSON schema with which the message field can be serialized. This field is used exclusively with the messageSchemaId field.
- **message** (bytes)
An Avro encoded message, which is the payload of the message envelope.

Given the envelope schema, it is possible for anyone using this schema to properly decode the envelope messages from Microsoft Azure Event Hubs.

Once the envelope message has been decoded, there are two possible scenarios:

- **Scenario 1:** Decoding a self-describing message such as the metadata message
- **Scenario 2:** Decoding a message by referenced schema ID such as data messages

The method for logically decoding messages in both scenarios is described below.

Decoding a self-describing message

When the messageSchema field is not null, it means the message field can be decoded using the schema included in the messageSchema field. This is fairly straightforward to perform programmatically since the only thing you need to usually supply Avro is a schema and a message, both of which are provided in the envelope message.

The Replicate metadata messages which include both table metadata, lineage and data schema description (to be referenced later by data messages) are enveloped in the self-describing envelope.

Decoding a message by referenced schema ID

Avro schemas are JSON documents which can be quite large, usually much larger than the data encoded by Avro conforming to the schema. For example, a schema of a 10 column table could be a JSON document of more than 100 characters while an actual row encoding of 10 columns may be only 10 bytes (depending of course on the type and length of fields). It is therefore typically not recommended to include schema and data together in a Microsoft Azure Event Hubs message because the schema information is redundant and is the same for all data messages while the actual data is the only thing which differs between data messages.

To avoid sending schema with each data message, each schema has a 32 bytes long ID. When a data message based on a previously sent data message schema (via the metadata message) is constructed, the `messageSchema` field is set to null and the `messageSchemaId` field is set to the 32 bytes ID of the schema instead. The application responsibility is to locate the data schema sent earlier in the metadata message and use that schema to decode the data message contained in the `message` field.

Typical consumer logic

A typical scenario involving Microsoft Azure Event Hubs involves Qlik Replicate as the Producer of messages into Microsoft Azure Event Hubs and customer code as the Consumer. Qlik Replicate offers the ability to define a specific hub as the schema hub and different hubs for the table data.

The customer's consumer code should read metadata messages from the schema hub and then save the data schemas and any other information the consumer wishes to access later in a customer defined zone. Another set of customer consumers should read data messages from the various data hubs, and access the data schemas zone as required to retrieve the data schemas required for decoding the data messages.

When consuming data messages and metadata messages from several hubs and partitions in a multi-thread/process manner, a situation may arise where a given consumer may attempt to read a data message *before* the corresponding metadata message has been read. As it is not possible to read a data message before its corresponding metadata message, the consumer's logic should wait a reasonable amount of time until the corresponding metadata message has been read. If the metadata message is still not available after waiting for a reasonable amount of time, the consumer should handle this as an unexpected error and activate the planned error policy. An example of such a policy could be saving the message in a dedicated "delayed" hub for later processing.

As a rule of thumb, the number of metadata messages will be much lower (in the magnitude of 1:10000 or more) than the number of data messages. So, assuming a metadata consumer is active, the gap between metadata message and data message should be no more than a few seconds (usually, milliseconds).

Metadata and data messages

This topic describes the structure and content of the metadata and data messages produced by the Microsoft Azure Event Hubs target endpoint.

Metadata message

Metadata message

Field	Type	Description
<code>schemald</code>	String	The unique identifier of the Avro schema.

10 Adding and managing target endpoints

Field	Type	Description
lineage	Structure	Information about the origin of the data (Replicate server, task, table, and so on)
server	String	The name of the Replicate server.
task	String	The name of the task.
schema (Schema Name)	String	The name of the database schema.
table (Table Name)	String	The name of the table.
tableVersion	Integer	Replicate maintains a version number of the structure of source table. Upon DDL change on the source, the version is increased and a new metadata message is produced.
timestamp	String	The date and time of the metadata message.
tableStructure	Structure	Describes the structure of the table.
tableColumns	Structure	Contains the list of columns and their properties.
{columns}	Structure	For each column, a record with the below properties.
ordinal	Integer	The position of the column in the record.
type	String	The column data type.
length	Integer	The maximum size of the data (in bytes) permitted for the column.
precision	Integer	For NUMERIC data type, the maximum number of digits required to represent the value.
scale	Integer	For NUMERIC data type, the maximum number of digits to the right of the decimal point permitted for a number.
primaryKeyPosition	Integer	The position of the column in the table's Primary Key. or Unique Index. The value is zero if the column is not part of the table's Primary Key.
dataSchema	String	The Avro schema for deserializing the Data messages.

Data message


Data message

Field	Type	Description
schema (name)	String	The name of the source database schema containing the replicated source table(s).
table (name)	String	The name of the source table.
headers	Structure	Information about the current record.


10 Adding and managing target endpoints

Field	Type	Description
operation (Operation)	Enum	The operation type. Full Load - Initial load of the source data to the target topic(s) REFRESH - Insertion of a record to the target during Full Load CDC - Applies source table changes to the relevant topic INSERT - Insertion of new target record UPDATE - Update of an existing target record DELETE - Deletion of a target record
changeSequence (Change Sequence)	String	A monotonically increasing change sequencer that is common to all change tables of a task. Use this field to order the records in chronological order. Applicable to CDC operations only.
timestamp (Timestamp)	String	The original change UTC timestamp. Applicable to CDC operations only.
streamPosition	String	The source CDC stream position. Applicable to CDC operations only.
transactionId (Transaction ID)	String	The ID of the transaction that the change record belongs to. Use this field to gather all changes of a specific transaction. Applicable to CDC operations only.

10 Adding and managing target endpoints

Field	Type	Description
changeMask (Change Mask)	String	<p>Indicates which data columns were changed in the source table.</p> <p>The change mask is a string of hexadecimal digits, representing a bitmask of data columns in little-endian order. The bit position in the change mask is based on the ordinal of the column in the metadata message of that table.</p> <p>This means that if there are 10 data columns, they occupy bits 0 to 9 in the bitmask.</p> <p>If UPDATE mask is 0B hexadecimal, which is 1011 binary – it means that the columns at ordinals 1, 2 and 4 were changed.</p> <p>The following describes the bit semantics:</p> <ul style="list-style-type: none">• For INSERT records, all the inserted columns have the associated bits set.• For DELETE records, only primary-key (or unique index) columns have the associated bits set. This allows an applier to construct a DELETE statement without having to find the primary key fields from another source.• For UPDATE records, each column with a changed value will have the associated bit set. <div data-bbox="699 1205 1390 1339" style="border: 1px solid #ccc; padding: 5px;"><p> <i>LOB columns are not included in the changeMask bit.</i></p></div>

10 Adding and managing target endpoints

Field	Type	Description
columnMask (Column Mask)	String	<p>Indicates which data columns are present in the message. Usually, this will include all of the table columns.</p> <div style="border: 1px solid gray; padding: 5px; margin: 10px 0;">  <i>When replicating from an Oracle source without full supplemental logging, some columns might not be present in the data, since they could not be replicated.</i> </div> <p>The column mask is a string of hexadecimal digits, representing a bitmask of data columns in little-endian order. The bit position in the column mask is based on the ordinal of the column in the metadata message for that table.</p> <p>This allows the applier to distinguish a null value that is the <i>actual</i> value of the column, from a null value that represents a column which could not be replicated from the source database.</p>
externalSchemald	String	<p>The Schema ID. This will only be displayed if the Include external Schema ID header check box is selected.</p> <p>As the Schema ID changes whenever a DDL is performed on the source table, consumer applications can use this information to determine if the message schema has changed.</p>
transactionEventCounter (Transaction Event Counter)	Long	<p>The sequence number of the current operation in the transaction.</p> <p>This can be used to determine the order of operations within a transaction.</p>
transactionLastEvent (Transaction Last Event)	Boolean	<p>"True" indicates that it is the final record in the transaction whereas "False" indicates that not all of the records have been processed.</p>
data	Structure	The data of the table record
{columns}		The column names and values in the current record.
beforeData	Structure	The data of the table record, before the change
{columns}		<p>The column names and values, before the change.</p> <p>Applicable to UPDATE operation.</p>

10.31 Using Microsoft Azure HDInsight as a target

This section describes how to set up and use Microsoft Azure HDInsight as the target endpoint in a replication task.

In this section:

- *Prerequisites (page 913)*
- *Limitations (page 913)*
- *Change data partitioning on Microsoft Azure HDInsight (page 914)*
- *Microsoft Azure HDInsight endpoint target data types (page 914)*
- *Setting general connection properties (page 915)*
- *Setting advanced connection properties (page 918)*

Prerequisites

Before you begin to work with Microsoft Azure HDInsight as a target in Qlik Replicate, make sure that the following prerequisites have been met:

- **General:**
 - The ADLS or Blob storage location (whichever you are using) must be accessible from the Qlik Replicate machine.
 - The user specified in the Microsoft Azure HDInsight target endpoint's [Hive access settings](#) must have access to HiveServer2.
- **ODBC Driver when Replicate Server is running on Windows:** Microsoft Hive ODBC Driver 2.06.12.1012 or later must be installed on the Qlik Replicate Server machine.
- **ODBC Driver when Replicate Server is running on Linux:**
 1. Install Cloudera ODBC driver for Apache Hive 2.06.09.1009 or later on the Replicate Server machine.
 2. After the driver is installed, edit the **hortonworks.hiveodbc.ini** file as follows:

```
DriverManagerEncoding=UTF-16
ODBCInstLib=libodbcinst.so
```
- **Permissions:**
 - The "Storage account" (when using Blob storage) or "Azure Active Directory application ID" (when using ADLS) specified in the Microsoft Azure HDInsight endpoint's **Storage** settings must have write access to the specified Blob/ADLS storage target folder.
 - In the Access Control (IAM) settings for the ADLS Gen2 file system, assign the "Storage Blob Data Contributor" role to Replicate (AD App ID). It may take a few minutes for the role to take effect.
 - **Hive table permissions:** Replicate requires permissions to perform the following operations on Hive tables: CREATE, DROP, DESCRIBE, and ALTER table. ALTER table may also include RENAME table, ADD/RENAME column, and CREATE/DROP partitions on tables.

Supported blob storage types

The following blob storage types are supported:

- Standard storage with block blobs
- Premium storage with block blobs only

Limitations

The following limitations apply:

- UPDATE/DELETE DMLs are not supported during change processing. If an UPDATE/DELETE DML was captured on the source, it will be ignored on the target and a warning will be written to the log. If the Store Changes option is enabled in the task settings, these records will be written to the Change Table.
- Limited LOB support only.
- Dropping columns and changing column data types or the data type length is not supported and will suspend the table in all cases *except* for data types that are mapped to STRING. Changes to the data

type length of a data type mapped to STRING (e.g. VARCHAR(50) to VARCHAR(100)) will simply be ignored.

- Writing to "special" tables such as tables with partitions, buckets, or skews is not supported.
- The following Control Tables are not supported as they require UPDATE/DELETE operations, which are not supported:
 - Replication Status (requires UPDATE).
Name on target: `attrep_status`
 - Suspended Tables (requires DELETE).
Name on target: `attrep_suspended_tables`
For more information on Control Tables, see *Control tables (page 1169)*.
- Table and column names can only contain ASCII characters.
- Column names cannot contain the '?' symbol. If needed, the '?' symbol can be replaced with a valid ASCII character [using a transformation](#).
- Blob storage limitations:
 - Supported on Windows only
 - Append operation is not supported
- Proxy:
 - Does not affect the ODBC (Hive) connection. Affects the storage connection only.
 - When using Blob storage, the HTTPS scheme is not supported.
 - Accessing Azure Data Lake Storage (ADLS) Gen2 storage via a proxy is not supported with HTTPS protocol.
 - Connecting to a proxy server with a username and password is not supported with [Azure Data Lake Storage \(ADLS\) Gen2 storage](#).

Change data partitioning on Microsoft Azure HDInsight

When *Change Data Partitioning (page 1177)* is enabled, the Replicate Change Tables in Hive are partitioned by the `partition_name` column. Data files are uploaded to the Blob/ADLS storage, according to the [maximum size and time definition](#), and then stored in a directory under the Change Table directory. Whenever the specified partition timeframe ends, a partition is created in Hive, pointing to the Blob/ADLS storage.

Information about the partitions is written to the [attrep_cdc_partitions](#) Control Table.

Microsoft Azure HDInsight endpoint target data types

The following table shows the Microsoft Azure HDInsight endpoint target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

Qlik Replicate and Microsoft Azure HDInsight data type comparison

Qlik Replicate Data Types	Microsoft Azure HDInsight Data Types
BOOL	BOOLEAN
BYTES	STRING
TIME	STRING
DATETIME	TIMESTAMP
DATE	DATE
INT1	TINYINT
INT2	SMALLINT
INT4	INT
INT8	BIGINT
NUMERIC	DECIMAL (p,s)
REAL4	FLOAT
REAL8	DOUBLE
STRING	VARCHAR (Length)
UINT1	SMALLINT
UINT2	INT
UINT4	BIGINT
UINT8	DECIMAL (20,0)
WSTRING	VARCHAR (Length)
BLOB	STRING
NCLOB	STRING
CLOB	STRING

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a Microsoft Azure HDInsight target endpoint to Qlik Replicate:

1. In the Qlik Replicate console, click **Manage Endpoint Connections** to open the **Manage Endpoint Connections** dialog box.
For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints* (page 114).

2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the endpoint being used.
3. In the **Description** field, type a description that helps to identify the Microsoft Azure HDInsight endpoint. This is optional.
4. Select **Microsoft Azure HDInsight** as the endpoint **Type**.
5. In the **Security** section's **SSL CA path** field, specify the location of the SSL CA certificate *in PEM format* (or certificate chain if needed) on the Replicate Server machine. If you do not have your own CA certificate, you can specify the path to the `cacerts.pem` file provided with the ODBC driver installation.
Note that if you select the **Azure Data Lake Storage (ADLS) Gen1** storage type, the CA file must also contain the CA of `login.microsoftonline.com`.
6. In the **Azure Storage** section, from the **Storage type** drop-down list, select the desired storage type.



The **Blob storage** option is not supported when Qlik Replicate is running on Linux.

If you selected **Blob storage**:

- a. In the **Storage account** field, specify the name of an account with write permissions to the container.



To connect to an Azure resource on Government Cloud or China Cloud, you need to specify the full resource name of the storage account. For example, assuming the storage account is `MyBlobStorage`, then the resource name for China cloud would be `MyBlobStorage.dfs.core.chinacloudapi.cn`

For information on setting internal parameters, see *Setting advanced connection properties* (page 918)

- b. In the **Access key** field, specify the account access key.
- c. In the **Container name** field, specify the container name.
- d. In the **Target folder** field, specify where to create the data files on Blob storage.

If you selected **Azure Data Lake Storage (ADLS) Gen1**:

- a. In the **Data Lake Store name** field, specify the full name of the ADLS storage.



To connect to an Azure resource on Government Cloud or China Cloud, you need to specify the full resource name of the storage account. For example, assuming the storage account is "myaccount", then the resource name for China Cloud would be `myaccount.dfs.core.chinacloudapi.cn`

In addition, you also need to specify the login URL using the `adlsLoginUrl` internal parameter. For China Cloud, this would be `https://login.chinacloudapi.cn`

For information on setting internal parameters, see *Setting advanced connection properties* (page 918)

- b. In the **Azure Active Directory Tenant ID** field, specify the Azure Active Directory tenant ID.
c. In the **Application Registration Client ID** field, specify the application registration client ID.
d. In the **Application Registration Secret** field, specify the application registration secret.
e. In the **Target folder** field, specify where to create the data files on ADLS.

If you selected **Azure Data Lake Storage (ADLS) Gen2**:

- a. In the **Storage account** field, specify the name of your storage account.



To connect to an Azure resource on Government Cloud or China Cloud, you need to specify the full resource name of the storage account. For example, assuming the storage account is "myaccount", then the resource name for China Cloud would be `myaccount.dfs.core.chinacloudapi.cn`

In addition, you also need to specify the login URL using the `adlsLoginUrl` internal parameter. For China Cloud, this would be `https://login.chinacloudapi.cn`

For information on setting internal parameters, see *Setting advanced connection properties* (page 918)

- b. In the **Azure Active Directory Tenant ID** field, specify the Azure Active Directory tenant ID.
c. In the **Application Registration Client ID** field, specify the application registration client ID.
d. In the **Application Registration Secret** field, specify the application registration secret.
e. In the **Container** field, specify the file system containing your folders and files.
f. In the **Target folder** field, specify where to create the data files on ADLS.




- The **Target folder** name can only contain ASCII characters.
- Connecting to a proxy server with a username and password is not supported with [Azure Data Lake Storage \(ADLS\) Gen2 storage](#).

7. In the **Hive Access** section:
 - a. In the **Host** field, specify the host name of the Hive server.
 - b. Enter your **Username** and **Password** for accessing the Hive server in the designated fields.
 - c. In the **Database** field, specify the name of the Hive target database.


Setting advanced connection properties


The table below describes the settings in the **Advanced** tab.

Advanced tab settings

Setting	Description
File Format	Expand this section to specify or view the file format settings.
Target storage format	<p>Select either Text or Sequence.</p> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> <i>When using Text as the target storage format, new lines as part of the data values are not supported (due to an Apache Hadoop limitation). Best practice is to use Sequence as the target storage format.</i></p> </div>
Field delimiter	The delimiter that will be used to separate fields in the target file. The default is \001. Note that field delimiters should be ASCII characters unless they are non-printable control characters, in which case they should be in Octal format (e.g. \247).
Null value	<p>The value that will be used to indicate a null value in the target file.</p> <p>Example (where @ is the null value):</p> <p>mike,male,295678</p> <p>sara,female,@</p>
Escape character	<p>The escape character is used to escape the field delimiter character. When a field delimiter is escaped, it is interpreted as actual data, and not as a field delimiter.</p> <p>Example (where \ is the escape character and a comma is the field delimiter):</p> <p>sunroof\,power-steering</p>

10 Adding and managing target endpoints

Setting	Description
Add metadata header	<p>When the target storage format is set to Text, you can optionally add a header row to the data files. The header row can contain the source column names and/or the intermediate (i.e. Replicate) data types.</p> <p>Example of a target file with a header row when both With column names and With data types are selected:</p> <pre>Position:DECIMAL(38,0),Color:VARCHAR(10) 1,"BLUE" 2,"BROWN" 3,"RED" ...</pre>
File Attributes	Expand this section to specify or view the file attributes.
Maximum file size	Specify the maximum file size of each target file. When the data reaches the maximum size, the file will be closed and written to the specified target folder.
Compress files using	Select the compression method to use on ADLS/Blob storage.
Change Processing	Expand this section to specify or view change processing settings.
Apply/Store changes when file size reaches	Specify the minimum size of the data required to create a file in idle state.
Apply/Store changes when elapsed time reaches	Specify the maximum time to wait before applying the changes in idle state.
Proxy Server	<p>Expand this section to specify or view proxy settings.</p> <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;">  <i>Connecting to a proxy server with a username and password is not supported with Azure Data Lake Storage (ADLS) Gen2 storage.</i> </div>
Use proxy server	Select this option to access Microsoft Azure HDInsight via a proxy server.
Host name	The host name of the proxy server.
Port	The port via which to access the proxy server.

Setting	Description
User name	The user name for accessing the proxy server.
Password	The password for accessing the proxy server.
Scheme	Select which protocol to use to access the server (HTTP or HTTPS). When using Blob storage, only HTTP is available. When using ADLS, both HTTP and HTTPS are available. <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <i>The proxy configuration applies to the storage (Blob/ADLS), but does not apply to the ODBC connection.</i></div>
SSL CA Path:	The location of the CA certificate on the Replicate Server machine when HTTPS is the selected Scheme .



*To facilitate rapid delivery of DDL messages, files are uploaded immediately, regardless of the specified **File size reaches** or **Elapsed time reaches** values.*

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.32 Using Microsoft Azure Synapse Analytics as a target

This section describes how to set up and use Microsoft Azure Synapse Analytics as a target in a replication task. Microsoft Azure Synapse Analytics is located in the cloud and is accessed through your Microsoft Azure account.

In this section:

- *Prerequisites (page 922)*
- *Limitations and considerations (page 924)*
- *Supported data types (page 925)*
- *Setting general connection properties (page 926)*
- *Setting advanced connection properties (page 929)*

Prerequisites

The following sections describe the prerequisites necessary for using Microsoft Azure Synapse Analytics as a target endpoint in a Replicate task.

- *Sign up for Microsoft Azure Blob Storage (not required when using ADLS Gen2 Storage) (page 922)*
- *Sign up for Microsoft Azure Synapse Analytics (page 922)*
- *Install the required client (page 922)*
- *Required permissions (page 923)*
- *Open the required firewall port(s) (page 924)*

Sign up for Microsoft Azure Blob Storage (not required when using ADLS Gen2 Storage)

Sign up for an Azure Blob Storage account and make a note of the account name, account key, container name and target folder - you will need to provide them later.



For best performance, the Azure Blob Storage container should be in the same region as your Microsoft Azure Synapse Analytics .

Supported blob storage types

The following blob storage types are supported:

- Standard storage with block blobs
- Premium storage with block blobs only

Sign up for Microsoft Azure Synapse Analytics

Sign up for Microsoft Azure Synapse Analytics and make a note of the server name, port, user name, password, database name and Azure Blob Storage access Credential - you will need to provide them later. Note that if you have not already created an Azure Blob Storage access Credential, you can configure Replicate to create one automatically as described in *Setting general connection properties (page 926)*.

Install the required client

Install one of the supported clients, as described below.

Qlik Replicate for Windows

For all versions of the source database, install Microsoft ODBC Driver 17.6 for SQL Server or later on the Qlik Replicate Server machine.

Qlik Replicate for Linux

1. Install Microsoft ODBC Driver 13 (13.01.0009) for SQL Server or later on the Replicate machine.
2. Change the working directory to `<product_dir>/bin`.
3. Stop the Replicate service:
`./areplicate stop`
4. Optionally, confirm that the service has stopped:

```
./areplicate status
```

5. Copy the driver location to the `site_arep_login.sh` file as follows:

```
echo "export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/microsoft/msodbcsql<version>/lib64/"
>> site_arep_login.sh
```

Example:

```
echo "export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/microsoft/msodbcsql17/lib64/" >>
site_arep_login.sh
```

This will add the driver to "LD_LIBRARY_PATH" and update the driver location in the **site_arep_login.sh** file.

6. Optionally, confirm that the driver location was copied:

```
cat site_arep_login.sh
```

7. Start the Replicate service:

```
./areplicate start
```

8. Optionally confirm that the service has started:

```
./areplicate status
```

Required permissions

The following topic describes which permissions are required when using Microsoft Azure Synapse Analytics as a target endpoint in a Replicate task.

Storage permissions

The user specified in the Microsoft Azure Synapse Analytics endpoint settings must be granted the following storage permissions.

- **Applicable when using either Blob storage or ADLS Gen2 storage:**

- On the Azure Blob/ADLS Gen2 Storage container: LIST
- On the Azure Blob/ADLS Gen2 Storage folder: READ, WRITE and DELETE
- When electing to use the **COPY Statement** load method (in the endpoint settings' **General** tab), Synapse uses its Managed Identity to access the storage account. This requires the Synapse identity to be granted the "Storage Blob Data Contributor" permission on the storage account.

- **Relevant when using ADLS Gen2 storage only:**

- In the Access Control (IAM) settings for the ADLS Gen2 file system, assign the "Storage Blob Data Contributor" role to Replicate (AD App ID). It may take a few minutes for the role to take effect.

Database permissions

The user specified in the Microsoft Azure Synapse Analytics endpoint settings must be granted the following permissions:

- **General permissions:**

- db_owner role

- **Table permissions:**

- SELECT, INSERT, UPDATE and DELETE
- Bulk Load
- CREATE, ALTER, DROP (if required by the task's definition)

Open the required firewall port(s)

- **When Replicate runs on a machine outside Azure** - Open port 1433 for outbound communication.
- **When Replicate runs on an AzureVM** - Open the following ports for outbound communication:
 - 1433
 - 11000-11999
 - 14000-14999

Limitations and considerations

The following section describes the limitations of using Microsoft Azure Synapse Analytics as a Replicate target.

- Source columns with CHAR/VARCHAR data types and a non-Latin collation (e.g. "Chinese_PRC_CI_AS") need to be mapped to NVARCHAR. This can be done by defining a global transformation for all tables in the replication task or by defining a single transformation for a specific table. For more information on defining transformations, see *Starting the Global Transformation Rules wizard (page 1107)* and *Defining transformations for a single table/view (page 1084)*.
- Empty source column handling:
 - When using the **Polybase** load method, an empty value in a source text column (for example VARCHAR or CLOB) will be inserted as a space into the corresponding target column. Therefore, if you need empty values in source text columns to be inserted as empty values into the corresponding target columns, use the **COPY Statement** load method instead. For more information on the loading options, see *Setting general connection properties (page 926)*.
 - When replicating a source binary column (for example, VARBINARY or BLOB) with an empty value, Replicate inserts a NULL into the corresponding target column.



Empty values are not NULL, but data with zero length.

- The rename column DDL is not supported.
- Blob storage is not supported when Qlik Replicate is installed on Linux.
- Full LOB replication is not supported.
- Replication of records exceeding 1 MB is not supported.
- Replication of data with 4-byte emoji characters requires the data type to be a wide string (for example, NCHAR/NVARCHAR/NCLOB) on the target.
- When **PolyBase** is selected as the **Load data using** method, the characters **\r** and **\n** in the data are replaced with a space. For new line support, you must select **COPY Statement** as the **Load data using** method.
See also: *Setting general connection properties (page 926)*.

Change processing (CDC) limitations

- Capture of BLOB column changes exceeding 2000 bytes is not supported
- Capture of CLOB/NCLOB column changes exceeding 4000 bytes is not supported

Supported data types

The Microsoft Azure Synapse Analytics Replicate endpoint supports most Microsoft Azure Synapse Analytics data types. The following table shows the Microsoft Azure Synapse Analytics target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source database you are using. For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).

Supported data types

Qlik Replicate Data Types	Microsoft Azure Synapse Analytics Data Types
BOOL	BIT
BYTES	If length is => 1 and =< 8000, then: VARBINARY (Length in Bytes) If length is => 8001 and =< 2147483647, then: VARBINARY (8000)
DATE	DATE
TIME	TIME
DATETIME	DATETIME2 (s)
INT1	SMALLINT
INT2	SMALLINT
INT4	INTEGER
INT8	BIGINT
NUMERIC	DECIMAL (p,s)
REAL4	FLOAT(24)
REAL8	FLOAT(53)
STRING	If length is => 1 and =< 8000, then: VARCHAR (Length in Bytes) If length is => 8001 and =< 2147483647, then: VARCHAR (8000)
UINT1	TINYINT
UINT2	SMALLINT
UINT4	INTEGER
UINT8	BIGINT

Qlik Replicate Data Types	Microsoft Azure Synapse Analytics Data Types
WSTRING	If length is => 1 and =< 4000, then: NVARCHAR (Length in Bytes) If length is => 4001 and =< 2147483647, then: NVARCHAR (4000)
BLOB	Full Load: VARBINARY (8000) Change Processing: VARBINARY (4000)
NCLOB	NVARCHAR (4000)
CLOB	Full Load: VARCHAR (8000) Change Processing: VARCHAR (4000)



*About Microsoft Azure Synapse Analytics LOB support:
Full LOB data types are not supported. For information on including Limited-size LOB data types in the replication, see the Metadata (page 1166) tab description in Customizing tasks (page 1082).*

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a Microsoft Azure Synapse Analytics target to Qlik Replicate:

1. In the Qlik Replicate Console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box.
2. In the **Manage Endpoint Connections** dialog box, click **New Endpoint Connection**.
3. In the **Name** field, type a name for your Microsoft Azure Synapse Analytics Synapse Analytics [service]. This can be any name that will help to identify your Microsoft Azure Synapse Analytics database.
4. In the **Description** field, type a description that helps to identify the Microsoft Azure Synapse Analytics target database. This is optional.
5. Select **Target** as the **role**.
6. Select **Microsoft Azure Synapse Analytics** as the **Type**.
7. Configure the settings as described below.

Synapse ODBC Access

- **Server name:** The name of the Microsoft Azure Synapse Analytics server you are using.
- **Authentication method:** Select either **SQL Authentication** or **Azure Active Directory Authentication** as appropriate.
- **Port:** The port number for connecting to Microsoft Azure Synapse Analytics .
- **User name:** The user name of a registered Microsoft Azure Synapse Analytics user.
- **Password:** The password for the user entered in the **User name** field.
- **Database name:** The target database name.
If you do not have these values, contact the Microsoft Azure account owner or your company's Microsoft Azure Synapse Analytics System Administrator.

Staging on Azure

In the **Staging on Azure** section, from the **Storage type** drop-down list, select either **Microsoft Azure Blob Storage** or **Microsoft Azure Data Lake Storage (ADLS) Gen2** according to your storage type.

Microsoft Azure Blob Storage



The **Blob storage** option is not supported when Qlik Replicate is running on Linux.

- **Azure Storage account name:** The name of an account with write permissions to the container.



To connect to an Azure resource on Government Cloud or China Cloud, you need to specify the full resource name of the storage account. For example, assuming the storage account is `MyBlobStorage`, then the resource name for China cloud would be `MyBlobStorage.dfs.core.chinacloudapi.cn`

For information on setting internal parameters, see *Setting advanced connection properties* (page 918)

- **Container name:** The container name.



Container names must comply with the Microsoft guidelines. For more information, see the *Microsoft Help* on this subject.

- **Access key:** The account access key.
- **Staging folder:** The folder in which to create the data files on Blob storage.

Microsoft Azure Data Lake Storage (ADLS) Gen2

- **Storage account** field, specify the name of your storage account.



To connect to an Azure resource on Government Cloud or China Cloud, you need to specify the full resource name of the storage account. For example, assuming the storage account is "myaccount", then the resource name for China Cloud would be `myaccount.dfs.core.chinacloudapi.cn`

In addition, you also need to specify the login URL using the `adlsLoginUrl` internal parameter. For China Cloud, this would be `https://login.chinacloudapi.cn`

For information on setting internal parameters, see *Setting advanced connection properties* (page 918)

- **Container:** The container containing your folders and files.
- **Azure Active Directory Tenant ID:** The Azure Active Directory tenant ID.
- **Application Registration Client ID:** The application registration client ID.
- **Application Registration Secret:** The application registration secret.
- **Staging folder:** The folder in which to create data files on ADLS.

Data Loading

- **Load data using:** Select **PolyBase** (Default) or **Copy Statement** according to the method you want Replicate to use to load your data into the target tables. Generally speaking, **Copy Statement** should provide better performance, but this might vary between specific environments.



- With the **COPY Statement** load method, Synapse uses its Managed Identity to access the storage account. This requires the Synapse identity to be granted the "Storage Blob Data Contributor" permission on the storage account.
- When **PolyBase** is selected as the **Load data using** method, the characters `\r` and `\n` in the data are replaced with a space. For new line support, you must select **COPY Statement** as the **Load data using** method.

- When the data is loaded using **PolyBase**, Microsoft Azure Synapse Analytics authenticates itself to the external data source using an SQL Server Credential. You can either configure Replicate to create the credential automatically during runtime (the default) or use an existing credential. If you selected **PolyBase**, select one of the following **Create SQL Server credential** options:
 - **Automatically:** Specify your **Access key**.
 - **Use existing credential:** Specify an existing **Credential name**.
- **Max file size (MB):** Relevant for Full Load and CDC. The maximum size a file can reach before it is loaded/applied to the target. If you encounter performance issues, try adjusting this parameter.



When using compression, the actual file size will be smaller.

- **Compress files before upload:** Compressing files might improve loading performance in certain environments, although it will also consume greater resources. Select this option to compress files before loading.
- **Number of files to apply in a batch:** Relevant for Full Load only. The number of files to apply in a single batch. If you encounter performance issues, try adjusting this parameter.
- **Batch load timeout (seconds):** If you encounter frequent timeouts when loading the files, try increasing this value.

Setting advanced connection properties

In the **Advanced** tab, you can set the following properties:

ODBC attributes

- **ODBC driver:** The name of the default ODBC driver you are using to connect to Microsoft Azure Synapse Analytics. The default driver is **ODBC Driver 17 for SQL Server**.
- **Additional ODBC connection properties:** Enter additional ODBC connection properties, if required.

Proxy server

- **Use proxy server** - Select this option to connect via a proxy server, and then provide the proxy connection settings:
- Choose **Connect via proxy server to** options according to which resources you want to access via a proxy server:
 - **Staging storage**
 - **Azure Active Directory**



*Only available with the **Microsoft Azure Data Lake Storage (ADLS) Gen2** storage type.
For information on the available storage types, see [Setting general connection properties](#) (page 926)*

- **Proxy connection:** Choose one of the following:
 - **Use endpoint proxy setting:** Choose this option to use the proxy settings defined for this endpoint.
 - **Use default proxy settings:** Choose this option to use the default proxy settings defined for all endpoints. The default settings are defined in the Replicate server settings' [Default Proxy Server](#) tab.
- **Host name:** The host name of the proxy server.
- **Port:** The port via which to access the proxy server.
- **User name:** The user name for the proxy server.
- **Password:** The password for the proxy server.



*Proxy authentication is only supported with the **Microsoft Azure Blob Storage** storage type.
For information on the available storage types, see [Setting general connection properties](#)*



(page 926)



The HTTPS scheme is not supported.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.33 Using Microsoft Azure SQL Database as a target

This section describes how to set up a Microsoft Azure SQL Database as a target in a replication task.

In this section:

- *Prerequisites (page 932)*
- *Limitations (page 933)*
- *Permissions (page 933)*
- *Microsoft Azure SQL Database target data types (page 933)*
- *Setting general connection properties (page 935)*
- *Setting advanced connection properties (page 936)*

Prerequisites

Make sure that the following prerequisites have been met:

Qlik Replicate for Windows

Install Microsoft ODBC Driver 18.1 for Windows on the Qlik Replicate Server machine.

Qlik Replicate for Linux

The procedure below assumes that you have installed a single default instance of Replicate on Linux (areplicate). If you have [installed multiple instances](#), replace areplicate with the name of the instance running the task with a Microsoft SQL Server source. If several instances are running such as task, the procedure needs to be repeated for each instance.

Install Microsoft ODBC Driver 18.1 for Linux on the Replicate machine.

Then, open a Unix shell and do the following:

1. Change the working directory to `<Replicate-Install-Dir>/bin`.
2. Stop the Replicate service:

```
./areplicate stop
```
3. Optionally, confirm that the service has stopped:

```
./areplicate status
```
4. Copy the driver location to the `site_arep_login.sh` file as follows:

```
echo "export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/microsoft/msodbcsql18/lib64/" >> site_arep_login.sh
```

This will add the driver to "LD_LIBRARY_PATH" and update the driver location in the **site_arep_login.sh** file.
5. Optionally, confirm that the driver location was copied:

```
cat site_arep_login.sh
```
6. Start the Replicate service:

```
./areplicate start
```
7. Optionally confirm that the service has started:

```
./areplicate status
```
8. Replicate requires the following ODBC library: `libmsodbcsql-18.1.so.1.1`
To check which library version is currently installed Issue the following command:

```
ls /opt/microsoft/msodbcsql/lib64/
```

If the existing library has a different version number (e.g. `libmsodbcsql-18.0.so.1.1`), you need to create a symbolic link between the existing library and the required library.
To do this, issue the following commands:

```
cd /opt/microsoft/msodbcsql18/lib64/
ln -s existing_library_name libmsodbcsql-18.1.so.1.1
```

where *existing_library_name* is the name of the currently installed library (e.g. `libmsodbcsql-18.0.so.1.1`).

TLS 1.2 prerequisites

If your environment matches the following:

- Microsoft SQL Server 2014
- Microsoft ODBC Driver 18 installed on Red Hat 8.1 or later

Then, in order to work with TLS 1.2, you must install a specific Service Pack on the Windows machine where Microsoft SQL Server is installed.

For more information, see <https://support.microsoft.com/en-us/help/3135244/tls-1-2-support-for-microsoft-sql-server>.

Other prerequisites

- Ports:
 - **When Replicate Server runs on a machine outside Azure** - Open port 1433 for outbound traffic.
 - **When Replicate Server runs on an AzureVM** - Open the following ports for outbound traffic:
 - 1433
 - 11000-11999
 - 14000-14999
- A Microsoft Azure SQL Database account with the specific access privileges is required. See [Target Permissions](#) for more information.

Limitations

When using a Microsoft Azure SQL Database as a target in a Replicate task, The following imitations apply:

- When **Use BCP for loading tables** is selected in the **Advanced** tab (the default), unlimited LOB columns are not supported in **Batch optimized apply** change processing mode. You can work around this limitation by limiting LOB column size in the task settings, clearing the **Use BCP for loading tables** option or switching to **Transactional apply** mode.
- When the **Use BCP for loading tables** option is enabled in the **Advanced** tab, triggers are not executed.
- When both the **Batch optimized apply** and **Limit LOB size to** options are enabled, BLOB columns on the target will be truncated to a quarter of the size specified in the [Limit LOB size to](#) field.

Permissions

The following describes the security requirements for using Qlik Replicate with a Microsoft Azure SQL Database target.

The Qlik Replicate user (i.e. the user specified in the Microsoft Azure SQL Database endpoint settings) must have at least the db_owner user role on the Microsoft Azure SQL Database you are connecting to.

A Microsoft Azure SQL Database system administrator must provide this permission for all Qlik Replicate users.

Microsoft Azure SQL Database target data types

The Microsoft Azure SQL Database target for Qlik Replicate supports most Microsoft Azure SQL Database data types. The following table shows the Microsoft Azure SQL Database target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source database you are using.

10 Adding and managing target endpoints

For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).



When replicating to Microsoft SQL Server 2019, UTF-8 data can be transferred to VARCHAR, CHAR, and CLOB columns when the database collation is set to %collation%_UTF8 (for example, **Latin1_General_100_CI_AS_SC_UTF8**). When working with an earlier Microsoft SQL Server version or a different collation, use a transformation (global or table-specific) to transform the column data type to WSTRING.

See also *Using the Transform tab* (page 1086) and *Defining global rules* (page 1107).

Supported data types

Qlik Replicate Data Types	Microsoft Azure SQL Database Data Types
BOOLEAN	TINYINT
BYTES	VARBINARY (length)
DATE	DATE
TIME	TIME(0)
DATETIME	DATETIME2 (scale)
INT1	SMALLINT
INT2	SMALLINT
INT4	INT
INT8	BIGINT
NUMERIC	NUMERIC (p,s)
REAL4	REAL
REAL8	FLOAT
STRING	If column is date or time then: DATETIME2 If the column is <i>not</i> a date or time: VARCHAR (length)
UINT1	TINYINT
UINT2	SMALLINT
UINT4	INT
UINT8	BIGINT
WSTRING	NVARCHAR (length)

Qlik Replicate Data Types	Microsoft Azure SQL Database Data Types
BLOB	VARBINARY (max) IMAGE To use this data type with Qlik Replicate, you must enable the use of BLOBs for a specific task. During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.
CLOB	VARCHAR (max) TEXT To use this data type with Qlik Replicate, you must enable the use of CLOBs for a specific task. During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.
NCLOB	NVARCHAR (max) NTEXT To use this data type with Qlik Replicate, you must enable the use of NCLOBs for a specific task. During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a Microsoft Azure SQL Database target endpoint to Qlik Replicate:

1. In the Qlik Replicate Console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box and then click **New Endpoint Connection**. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Microsoft Azure SQL Database database. This is optional.
4. Select **Target** as the database **role**.
5. Select **Microsoft Azure SQL Database** as the database **Type**.

6. Select one of the following authentication methods:
 - SQL Authentication
 - Azure Active Directory Authentication
 - Azure Active Directory Service Principal
7. In the **Username** field, enter your user name for accessing the database. When **Azure Active Directory Service Principal** is selected, this is the Client ID.
8. In the **Password** field, enter your password for accessing the database. When **Azure Active Directory Service Principal** is selected, this is the Secret Key.



To ensure that the password provided to Replicate is never used by any other principal, Replicate can be configured to automatically replace the user-entered password with a strong random password. For more information, see [Configuring Replicate to Automatically Replace the User-Entered Password](#).



- The username and password are case sensitive.
- To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**. If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box. To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

9. Type the **Database name** or click **Browse** and select one from the list of available databases. This is the name of the database from where you are replicating the data.

Testing the connection

To check if the connection information you entered is correct, click **Test Connection**.

If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

Setting advanced connection properties

In the **Advanced** tab, you can set the following properties:

- **Use BCP for loading tables:** Select this to transfer data for full-load operations using BCP.



When the target table contains an identity column that does not exist in the source table, you must disable the **Use BCP for loading tables** option.

- **BCP packet size:** The maximum size of the packets (in bytes) used to transfer data using BCP.

- **Filegroup for Qlik Replicate internal tables:** Optionally, specify a filegroup for the Qlik Replicate internal tables. When the replication task starts, all of the internal Qlik Replicate [control tables](#) will be created in the specified filegroup.

The following is an example of a command for creating a filegroup:

```
ALTER database replicate
ADD FILEGROUP Test1FG1;
GO
ALTER database replicate
ADD FILE
(
NAME = test1dat5,
FILENAME = 'C:\temp\DATA\t1dat5.ndf',
SIZE = 5MB,
MAXSIZE = 100MB,
FILEGROWTH = 5MB
)
TO FILEGROUP Test1FG1;
GO
```

- **Additional ODBC connection properties:** Specify any additional ODBC connection parameters that you want to use.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.34 Using Microsoft SQL Server as a target

This section describes how to set up and use a Microsoft SQL Server database as a target in a replication task.



When configuring a task with a Microsoft SQL Server target endpoint, note the following:

- If the [Apply Changes](#) task option is enabled and the change processing mode is set to [Batch optimized apply](#), it is recommended to enable the **Apply batched changes to multiple tables concurrently** option in the Change Processing Tuning (page 1181) tab.
- When the **Apply batched changes to multiple tables concurrently** option is selected, the option to set a Global Error Handling policy will not be available. Also, some of the task-specific error handling defaults will be different.
For more information, see the **Apply batched changes to multiple tables concurrently** option in the Change Processing Tuning (page 1181) tab.

In this section:

- [Supported editions \(page 939\)](#)
- [Prerequisites \(page 939\)](#)
- [Limitation and considerations \(page 940\)](#)
- [Permissions \(page 941\)](#)
- [Microsoft SQL Server target data types \(page 941\)](#)
- [Setting general connection properties \(page 943\)](#)
- [Setting advanced connection properties \(page 944\)](#)

Supported editions

Qlik Replicate supports the following Microsoft SQL Server editions:

- Enterprise Edition
- Standard Edition
- Workgroup Edition
- Developer Edition
- SQL Server Web Edition

Prerequisites

Make sure the following prerequisites have been met:

Client prerequisites

Qlik Replicate for Windows

Install Microsoft ODBC Driver 18.1 for Windows on the Qlik Replicate Server machine.

Qlik Replicate for Linux

The procedure below assumes that you have installed a single default instance of Replicate on Linux (areplicate). If you have [installed multiple instances](#), replace areplicate with the name of the instance running the task with a Microsoft SQL Server source. If several instances are running such as task, the procedure needs to be repeated for each instance.

Install Microsoft ODBC Driver 18.1 for Linux on the Replicate machine.

Then, open a Unix shell and do the following:

1. Change the working directory to *<Replicate-Install-Dir>/bin*.
2. Stop the Replicate service:
`./areplicate stop`
3. Optionally, confirm that the service has stopped:
`./areplicate status`
4. Copy the driver location to the `site_arep_login.sh` file as follows:
`echo "export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/microsoft/msodbcsql18/lib64/" >> site_arep_login.sh`
This will add the driver to "LD_LIBRARY_PATH" and update the driver location in the **site_arep_login.sh** file.
5. Optionally, confirm that the driver location was copied:
`cat site_arep_login.sh`
6. Start the Replicate service:
`./areplicate start`
7. Optionally confirm that the service has started:
`./areplicate status`
8. Replicate requires the following ODBC library: `libmsodbcsql-18.1.so.1.1`
To check which library version is currently installed Issue the following command:
`ls /opt/microsoft/msodbcsql/lib64/`

If the existing library has a different version number (e.g. `libmsodbcsql-18.0.so.1.1`), you need to create a symbolic link between the existing library and the required library.

To do this, issue the following commands:

```
cd /opt/microsoft/msodbcsql18/lib64/  
ln -s existing_library_name libmsodbcsql-18.1.so.1.1
```

where *existing_library_name* is the name of the currently installed library (e.g. `libmsodbcsql-18.0.so.1.1`).

TLS 1.2 prerequisites

If your environment matches the following:

- Microsoft SQL Server 2014
- Microsoft ODBC Driver 18 installed on Red Hat 8.1 or later

Then, in order to work with TLS 1.2, you must install a specific Service Pack on the Windows machine where Microsoft SQL Server is installed.

For more information, see <https://support.microsoft.com/en-us/help/3135244/tls-1-2-support-for-microsoft-sql-server>.

Limitation and considerations

When using a Microsoft SQL Server database as a target in a Replicate task, The following imitations apply:

- When **Use BCP for loading tables** is selected in the endpoint settings' **Advanced** tab (the default), the following limitations apply:
 - Full-sized LOB columns are not supported in **Batch optimized apply** change processing mode. You can work around this limitation by [limiting LOB column size in the task settings](#), clearing the **Use BCP for loading tables** option, or switching to **Transactional apply** mode.
 - If following large object types are empty in the source database (0 bytes), they will not be empty after Full Load:
 - **BLOB** will be written as 0x0000 (2 bytes)
 - **TEXT** (VARCHAR(max)) will be written as 0X00 (1 byte)
 - **NTEXT** (NVARCHAR(max)) will be written as 0x0000 (2 bytes)



If these types are empty during Change Processing, they will be correctly written to the target as empty (0 bytes).

- Triggers are not executed.

For more information on the **Use BCP for loading tables** mode, see *Setting advanced connection properties* (page 944).

- When both the **Batch optimized apply** and **Limit LOB size to** options are enabled, BLOB and CLOB columns on the target will be truncated to a quarter of the size specified in the [Limit LOB size to](#) field.
- If you need to replicate 4-byte emoji characters, use a transformation to convert the data type from `WSTRING(n)` to `WSTRING (n*2)`.

Permissions

The following describes the security requirements for using Qlik Replicate with a Microsoft SQL Server target.

The Qlik Replicate user must have at least the `db_owner` user role on the Microsoft SQL Server database you are connecting to.

A Microsoft SQL Server system administrator must provide this permission for all Qlik Replicate users.

Microsoft SQL Server target data types

The Microsoft SQL Server target for Qlik Replicate supports most Microsoft SQL Server data types. The following table shows the Microsoft SQL Server target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source database you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.



*When replicating to Microsoft SQL Server 2019 or 2022, UTF-8 data can be transferred to VARCHAR, CHAR, and CLOB columns when the database collation is set to %collation%_UTF8 (for example, **Latin1_General_100_CI_AS_SC_UTF8**). When working with an earlier Microsoft SQL Server version or a different collation, use a transformation (global or table-specific) to transform the column data type to WSTRING.*

See also Using the Transform tab (page 1086) and Defining global rules (page 1107).

Supported target data types

Qlik Replicate Data Types	Microsoft SQL Server Data Types
BOOLEAN	TINYINT
BYTES	VARBINARY(length)
DATE	DATE
TIME	TIME(0)
DATETIME	DATETIME2(scale)
INT1	SMALLINT
INT2	SMALLINT
INT4	INT
INT8	BIGINT
NUMERIC	NUMERIC(p,s)

Qlik Replicate Data Types	Microsoft SQL Server Data Types
REAL4	REAL
REAL8	FLOAT
STRING	<p>If column is date or time then:</p> <p>DATETIME2</p> <p>If the column is <i>not</i> a date or time:</p> <p>VARCHAR(length)</p>
UINT1	INT2
UINT2	INT4
UINT4	INT8
UINT8	NUMERIC(20)
WSTRING	NVARCHAR(length)
BLOB	<p>VARBINARY(max)</p> <p>IMAGE</p> <p>To use this data type with Qlik Replicate, you must enable the use of BLOBs for a specific task.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p>
CLOB	<p>VARCHAR(max)</p> <p>TEXT</p> <p>To use this data type with Qlik Replicate, you must enable the use of CLOBs for a specific task.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p>
NCLOB	<p>NVARCHAR(max)</p> <p>NTEXT</p> <p>To use this data type with Qlik Replicate, you must enable the use of NCLOBs for a specific task.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p>

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a Microsoft SQL Server target endpoint to Qlik Replicate:

1. In the Qlik Replicate Console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box and then click **New Endpoint Connection**. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Microsoft SQL Server database. This is optional.
4. Select **Target** as the database **role**.
5. Select **Microsoft SQL Server** as the database **Type**.
6. In the **Server name** field, specify the host name or IP address of the computer with the Microsoft SQL Server instance containing the target database.



To override the default port, add the port to the server name, separated by a comma. For example, if the server name is `myserver.company.local` and the port is `3333`, then the server name should be written like this:
`myserver.company.local,3333`

7. Select **Windows authentication** (only relevant when Replicate is installed on Windows) or **SQL Server authentication**.

If you select **Windows authentication**, you will work with the user credentials for the Windows domain. This privilege must be configured in the Microsoft SQL Server database by the system administrator. Note that this option is not relevant when Replicate Server is running on Linux.



When using Windows authentication, make sure that the user account that is associated with the **Qlik Replicate Server** service has Network read and write permissions. This must be configured by a Windows system administrator.

See also *Working with Windows authentication (page 367)*.

If you select **SQL Server authentication**, type the Microsoft SQL Server authentication information (User name, Password) for the authorized user for this Microsoft SQL Server database. If you do not know this information, see the Microsoft SQL Server System Administrator.

To ensure that the password provided to Replicate is never used by any other principal, Replicate can be configured to automatically replace the user-entered password with a strong random password. For more information, see [Configuring Replicate to Automatically Replace the User-Entered Password](#).



This information is case sensitive.



Make sure that the Microsoft SQL Server user has the correct access privileges. For information on how to provide the required privileges, see [Permissions \(page 941\)](#).



To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**.

If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

8. Type the **Database name** or click **Browse** and select one from the list of available databases. This is the name of the database to where you are replicating the data.

Setting advanced connection properties

In the **Advanced** tab, you can set the following properties:

- **Use BCP for loading tables:** Select this to transfer data for full-load operations using BCP.



When the target table contains an identity column that does not exist in the source table, you must disable the **Use BCP for loading tables** option.

- **BCP packet size:** The maximum size of the packets (in bytes) used to transfer data using BCP.
- **Filegroup for Replicate internal tables:** Optionally, specify a filegroup for the Replicate internal tables. When the replication task starts, all Replicate's internal [Control Tables](#) will be created in the specified filegroup.

The following is an example of a command for creating a filegroup:

```
ALTER database replicate
ADD FILEGROUP Test1FG1;
GO
ALTER database replicate
ADD FILE
(
NAME = test1dat5,
FILENAME = 'C:\temp\DATA\t1dat5.ndf',
SIZE = 5MB,
MAXSIZE = 100MB,
FILEGROWTH = 5MB
)
TO FILEGROUP Test1FG1;
GO
```

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.35 Using a MySQL-based database as a target

This section describes how to set up and use a MySQL target endpoint in a replication task.

You need to configure a MySQL endpoint when replicating to any of the following databases:

- MySQL
- MariaDB
- Amazon Aurora (MySQL)
- Amazon RDS for MariaDB
- Amazon RDS for MySQL
- Microsoft Azure Database for MySQL - Flexible Server

In this section:

- *Prerequisites (page 947)*
- *Limitations and considerations (page 948)*
- *Permissions (page 948)*
- *Supported data types (page 949)*
- *Setting general connection properties (page 951)*
- *Setting advanced connection properties (page 952)*

Prerequisites

Before using MySQL as an endpoint in a Qlik Replicate task, make sure that the prerequisites outlined below have been met.

Driver setup

Installing the driver on Windows

Install MySQL ODBC Unicode Driver 64-bit version 8.0.32 on the Replicate machine.

Installing the driver on Linux

1. Stop the Replicate service:
`./areplicate stop`
2. Optionally, confirm that the service has stopped:
`./areplicate status`
3. Install MySQL ODBC Unicode Driver 64-bit version 8.0.32 on the Replicate machine.
4. Change the working directory to `<Replicate-Install-Dir>/bin`.
5. Copy the driver location to the `site_arep_login.sh` file:
`echo "export LD_LIBRARY_PATH=\$LD_LIBRARY_PATH:/usr/lib64" >> site_arep_login.sh`
This will add the driver to "LD_LIBRARY_PATH" and update the driver location in the **site_arep_login.sh** file.
6. Optionally, confirm that the driver location was copied:
`cat site_arep_login.sh`
7. Make sure that the `/etc/odbcinst.ini` file contains an entry for MySQL, as in the following example:

```
[MySQL ODBC 8.0 Unicode Driver]
Driver = /usr/lib64/libmyodbc8w.so
UsageCount = 1
```
8. Start the Replicate service:
`./areplicate start`
9. Optionally confirm that the service has started:
`./areplicate status`

General prerequisites

Make sure that the following prerequisites have been met:

- A MySQL account with the required *Permissions* (page 948).
- A MySQL database with the tables that you want to replicate should be accessible in your network.
The following MySQL editions are supported:
 - MySQL Community Edition
 - MySQL Standard Edition
 - MySQL Enterprise Edition
 - MySQL Cluster Carrier Grade Edition

Working with MySQL 8

To work with MySQL 8 as a target endpoint, you need to:

- Set the parameter `secure_file_priv = NULL` in the MySQL database.
- If you are using LOB columns, set the parameter `sort_buffer_size = 128M` in the MySQL database.

Working with Azure Database for MySQL - Flexible Server

When working with Azure Database for MySQL - Flexible Server, the `sql_generate_invisible_primary_key` parameter must be set to 'OFF'.

For more information, see [Limitations in Azure Database for MySQL - Flexible Server](#)

Replicating 4-byte UTF8 emojis

To support replication of 4-byte UTF8 emojis, the target schema character set must be set to `utf8mb4`.

Limitations and considerations

The following limitations apply:

- When replicating to the MariaDB target database, if the DATETIME value is set to zero, you need to define a transformation that replaces the DATETIME with valid values. For example:
`replace($datetime_column, '9999-12-31 23:59:59', '2000-01-01 22:00:00')`
- Due to the way MySQL operates, when loading data to a MySQL target during a Full Load task, duplicate key errors will not be reported to the logs.
- When updating a column's value to its existing value, a zero rows affected is returned from MySQL (unlike Oracle and Microsoft SQL Server that perform an update of one row). This generates an entry in the `attrep_apply_exceptions` control table and the following warning:
some changes from the source database had no impact when applied to the target database.
see `attrep_apply_exceptions` table for details.
- Due to an issue with MySQL 5.7 (which was resolved in version 8.0), Replicate cannot insert multibyte characters into TEXT columns when the Full Load uses CSV files. Consequently, if the target endpoint is MySQL 5.7 and a replicated table contains TEXT columns with UTF-8 multibyte characters, the target table might be empty.

Workaround:

In the **Advanced** tab, set the `loadUsingCSV` internal parameter to `FALSE`. Note that this might impact performance.

- If you need to replicate 4-byte emoji characters, the target schema character set must be set to `utf8mb4`.

Permissions

You must provide MySQL account access to the Qlik Replicate user. This user must have read/write privileges in the MySQL database.

Additionally, for each target schema involved in the replication, the Qlik Replicate user must have the following permissions:

- ALTER
- CREATE

- CREATE TEMPORARY TABLES
- DELETE
- DROP
- INSERT
- SELECT
- UPDATE

Supported data types

The following table shows the MySQL database target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

Supported data types

Qlik Replicate Data Types	MySQL Data Types
BOOL	BOOL
BYTES	<p>If length is => 1 and =< 8095, then:</p> <p>VARBINARY (Length)</p> <p>If length is => 8096 and =< 65535, then:</p> <p>BLOB</p> <p>If length is => 65536 and =< 16777215, then:</p> <p>MEDIUMBLOB</p> <p>If length is => 16777216 and =< 2147483647, then:</p> <p>LOB</p>
DATE	DATE
TIME	TIME
DATETIME	<p>If scale is => 0 and =< 6, then:</p> <p>DECIMAL (p,s)</p> <p>If scale is => 7 and =< 12, then:</p> <p>VARCHAR (37)</p>
INT1	TINYINT
INT2	SMALLINT
INT4	INTEGER

10 Adding and managing target endpoints

Qlik Replicate Data Types	MySQL Data Types
INT8	BIGINT
NUMERIC	If scale is => 0 and =< 30, then: DECIMAL (p,s) If scale is => 31 and =< 100, then: VARCHAR (45)
REAL4	FLOAT
REAL8	DOUBLE
STRING	If length is => 1 and =< 8095, then: VARCHAR (Length) If length is => 8096 and =< 65535, then: TEXT If length is => 65536 and =< 16777215, then: MEDIUMTEXT If length is => 16777216 and =< 2147483647, then: LONGTEXT
UINT1	UNSIGNED TINYINT
UINT2	UNSIGNED SMALLINT
UINT4	UNSIGNED INTEGER
UINT8	UNSIGNED BIGINT
WSTRING	If length is => 1 and =< 8095, then: VARCHAR (Length) If length is => 8096 and =< 65535, then: TEXT If length is => 65536 and =< 16777215, then: MEDIUMTEXT If length is => 16777216 and =< 2147483647, then: LONGTEXT

Qlik Replicate Data Types	MySQL Data Types
BLOB	<p>When the Allow unlimited LOB size option is enabled:</p> <ul style="list-style-type: none"> • LONGBLOB <p>When the Limit LOB size to option is enabled:</p> <ul style="list-style-type: none"> • If the specified value is 63 KB or smaller, then: BLOB • If the specified value is 64 KB or larger, then: LONGBLOB <p>For information on limiting LOB size, see <i>Target metadata (page 1166)</i>.</p>
NCLOB	<p>When the Allow unlimited LOB size option is enabled:</p> <ul style="list-style-type: none"> • TEXT <p>When the Limit LOB size to option is enabled:</p> <ul style="list-style-type: none"> • If the specified value is 63 KB or smaller, then: TEXT • If the specified value is 64 KB or larger, then: LONGTEXT <p>For information on limiting LOB size, see <i>Target metadata (page 1166)</i>.</p>
CLOB	<p>When the Allow unlimited LOB size option is enabled:</p> <ul style="list-style-type: none"> • TEXT CHARACTER SET UTF-16 <p>When the Limit LOB size to option is enabled:</p> <ul style="list-style-type: none"> • If the specified value is 63 KB or smaller, then: TEXT CHARACTER SET UTF-16 • If the specified value is 64 KB or larger, then: LONGTEXT CHARACTER SET UTF-16 <p>For information on limiting LOB size, see <i>Target metadata (page 1166)</i>.</p>

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a MySQL target endpoint to Qlik Replicate:

1. In the Qlik Replicate Console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.

2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the MySQL database. This is optional.
4. Select **Target** as the database **role**.
5. From the **Type** drop-down list, select **MySQL**.
6. In the **Server** field, enter the host name or IP address of the computer on which the MySQL database is installed.



Notes

- *This information is case sensitive.*
- *To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**. If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box. To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.*

7. Optionally, change the default port (3306).
8. Type the MySQL authentication information (**User Name**, **Password**) for the authorized user for this MySQL database. If you do not know this information, see your MySQL database Administrator (DBA).
9. Select one of the following **Load source schemas into** options:
 - **The following database** - When this option is selected, all source schemas will be loaded into the selected database.
 - **Multiple databases** - When this option is selected, each of the source schemas will be loaded into its corresponding database. Additionally, by default, all Control Tables will be replicated to a new schema called **attrep_control**. You can override the default schema name as described in *Control tables (page 1169)*

Setting advanced connection properties

In the **Advanced** tab, you can set the following properties:

- **Max file size (KB)**: Select or type the maximum size (in KB) of a CSV file before it is loaded into the MySQL target database. The default value is 32000 KB.
- **Use parallel loading**: Select this option to improve performance when loading data into the MySQL target database.
 - **Use the following number of threads**: Specify how many threads to use to load the data into the MySQL target database. Note that setting a large number of threads may have an adverse effect on database performance since a separate connection is required for each thread.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.36 Using Netezza Performance Server (NPS) as a target

This section describes how to set up and use an Netezza Performance Server (NPS) database as a target endpoint in a replication task.

You should configure the Netezza Performance Server (NPS) target endpoint when replicating to any of the following targets:

- Netezza Performance Server (NPS) (On-premises)
- Netezza Performance Server (NPS) (Cloud)

In this section:

- *Prerequisites (page 955)*
- *Limitations (page 955)*
- *Permissions (page 955)*
- *Netezza Performance Server (NPS) target data types (page 956)*
- *Setting general connection properties (page 957)*
- *Setting advanced connection properties (page 958)*

Prerequisites



- Qlik Replicate must be installed on any Windows computer in your network.
- An Netezza Performance Server (NPS) account with the required access privileges is required. See *Permissions* (page 955).

Prerequisites for replicating to Netezza Performance Server (NPS) on premises

Make sure the following prerequisites have been met:

- Netezza Performance Server (NPS) ODBC 64-bit client installed on the Qlik Replicate machine.
- Netezza Performance Server (NPS) Tools 7.0.4.2 or later installed on the Qlik Replicate machine. Make sure that the Windows PATH environment variable includes the **bin** folder of Netezza Performance Server (NPS) Tools (<Installation directory>\bin).

Prerequisites for replicating to Netezza Performance Server (NPS) Performance Server

Make sure the following prerequisites have been met:

- Netezza Performance Server (NPS) 11.02.02.04 ODBC 64-bit client or later installed on the Qlik Replicate machine.
- IBM nps-winclient-v11 (11.2.2.4) or later installed on the Qlik Replicate machine. Make sure that the Windows PATH environment variable includes the **bin** folder of nps-winclient (<installation directory>\bin).

Limitations

Using Netezza Performance Server (NPS) as a target database in a Qlik Replicate task is subject to the following limitations:

- The Netezza Performance Server (NPS) target database uses the Netezza Performance Server (NPS) NZLOAD utility, which does not support loading tables with non-Latin names (e.g. Chinese). If any of your source tables has a non-Latin name, you can map it to a table with a Latin name. For more information on mapping table names, see *Performing General tasks for a single table/view* (page 1083) and *Starting the Global Transformation Rules wizard* (page 1107).
- Full LOB data types are not supported. For information on including Limited-size LOB data types in the replication, see the *Metadata* (page 1166) tab section in *Customizing tasks* (page 1082). Note also that the size of a row in the Netezza Performance Server (NPS) database *cannot exceed 64KB*. This should be taken into consideration when specifying the maximum LOB size in the *Metadata* (page 1166) tab.

Permissions

The user specified in the endpoint settings must be granted access to the Netezza Performance Server (NPS) account as well as the privileges listed below.

Database privileges

- LIST on <database> to <Replicate user>
- SELECT on <database> to <Replicate user>

Table privileges

- CREATE TABLE to <Replicate user>
- LIST on TABLE to <Replicate user>

Schema privileges

- CREATE SCHEMA to <Replicate user>
- LIST on SCHEMA to <Replicate user>

View privileges

- SELECT on _T_DATABASE to <Replicate user>
- SELECT on _V_SCHEMA to <Replicate user>
- SELECT on _V_USER to <Replicate user>
- SELECT on _V_TABLE to <Replicate user>
- SELECT on _V_TABLE_DIST to <Replicate user>
- SELECT on _V_RELATION_KEYDATA to <Replicate user>
- LIST on _T_DATABASE to <Replicate user>
- LIST on _V_SCHEMA to <Replicate user>
- LIST on _V_USER to <Replicate user>
- LIST on _V_TABLE to <Replicate user>
- LIST on _V_TABLE_DIST to <Replicate user>
- LIST on _V_RELATION_KEYDATA to <Replicate user>

Netezza Performance Server (NPS) target data types

The Netezza Performance Server (NPS) database for Qlik Replicate supports most Netezza Performance Server (NPS) data types. The following table shows the Netezza Performance Server (NPS) target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.



*Netezza Performance Server (NPS) does not support applying changes to binary data types in **Batch optimized apply** mode. For more information on **Batch optimized apply** mode, see *Change Processing Tuning* (page 1181).*

For information on how to view the data type that is mapped from the source, see the section for the source database you are using.

For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).

10 Adding and managing target endpoints

Supported data types

Qlik Replicate Data Types	Netezza Performance Server (NPS) Data Types
BOOLEAN	BOOLEAN
BYTES	VARCHAR (Length in Bytes)
DATE	DATE
TIME	TIME
DATETIME	If scale < 7: TIMESTAMP If scale 7-9: CHARACTER VARYING(37)
INT1	BYTEINT
INT2	SMALLINT
INT4	INTEGER
INT8	BIGINT
NUMERIC	NUMERIC (p,s)
REAL4	REAL
REAL8	DOUBLE
STRING	VARCHAR (Length)
UINT1	SMALLINT
UINT2	INTEGER
UINT4	BIGINT
UINT8	BIGINT
WSTRING	NVARCHAR (Length)
BLOB	VARCHAR (64,000)
NCLOB	NVARCHAR (7,000)
CLOB	VARCHAR (64,000)

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add an Netezza Performance Server (NPS) target endpoint to Qlik Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button.

2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Netezza Performance Server (NPS) database. This is optional.
4. Select **Target** as the database **role**.
5. Select **Netezza Performance Server (NPS)** as the database **Type**.
6. In the **Server** field, enter the name of the Netezza Performance Server (NPS) server.
7. Optionally, change the default **Port** (5480).
8. Type the Netezza Performance Server (NPS) authentication information (**User Name**, **Password**) for the authorized user for this Netezza Performance Server (NPS) database. If you do not know this information, see your Netezza Performance Server (NPS) database Administrator (DBA).



- *This information is required. If you are using the **Advanced** tab to create a custom string, make sure to include the **User Name** and **Password** properties. See *Setting advanced connection properties* (page 958) for more information.*
- *This information is case sensitive.*
- *If you want to set custom properties for this database, see *Setting advanced connection properties* (page 958).*



*Make sure that the Netezza Performance Server (NPS) user entered in the Netezza Performance Server (NPS) Authentication section has the correct access privileges. For information on how to provide the required privileges, see *Permissions* (page 955).*

9. In the **Database name** field, enter the name of the Netezza Performance Server (NPS) database.

Setting advanced connection properties

In the **Advanced** tab, you can set the following parameters:

- **Max file size:** Select or type the maximum size (in KB) of a CSV file before the file is loaded into the Netezza Performance Server (NPS) database. The default value is 32000 KB.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.37 Using ODBC to connect to a target

This topic describes how to use ODBC connectivity to connect to a target endpoint.



The ODBC target endpoint can be used to access targets that are not included in Qlik Replicate's extensive Support Matrix. However, unless enabled through Professional Services and approved by your Account Manager, the ODBC target endpoint should not be used. Additionally, the ODBC target endpoint should not be used to access any targets already listed in the Support Matrix.



When using HP NonStop SQL/MP (ARC) as an ODBC target, several additional procedures must be performed. For a detailed explanation, see [Using HP NonStop SQL/MP as an ODBC target](#) (page 1305).

In this section:

- [Prerequisites](#) (page 398)
- [ODBC target data types](#) (page 962)
- [Setting general connection properties](#) (page 963)
- [Setting advanced connection properties](#) (page 964)

Prerequisites

The following section describes the prerequisites for working with Qlik Replicate and an ODBC target endpoint.

In this section:

- *Qlik Replicate Server for Windows (page 961)*
- *Qlik Replicate Server for Linux (page 961)*

Qlik Replicate Server for Windows

You can connect an endpoint to Qlik Replicate using ODBC by indicating the DSN (Data Source Name). In this case you must be sure that a DSN is defined for the ODBC endpoint on the computer where Qlik Replicate is installed.

1. Install an endpoint client on the computer where Qlik Replicate is installed. The client you install depends on the ODBC provider you are using. For example, if you are using an IBM DB2 endpoint, install an IBM DB2 client.



You must use a 64-bit ODBC provider client to work with Qlik Replicate.

2. Use the ODBC Data Source Administrator to create a System DSN. The Data Source is located in the Windows control panel.

Qlik Replicate Server for Linux

The following section describes the steps you need to perform to work with Qlik Replicate for Linux and ODBC as a target endpoint in a Qlik Replicate task.

1. On the Qlik Replicate Server machine, install the ODBC client that you want to use (e.g. postgresQL).
2. Make sure that the `/etc/odbcinst.ini` file contains the correct entry for the driver you installed, as in the following example:

```
[PostgreSQL]
Description = ODBC for PostgreSQL
Driver = /usr/lib/psqlodbc.so
Setup = /usr/lib/libodbcpsqlS.so
Driver64 = /usr/lib64/psqlodbc.so
Setup64 = /usr/lib64/libodbcpsqlS.so
FileUsage = 1
```



To access an IBM DB2 for LUW target using ODBC, make sure that you specify the `libdb2o.so` driver (and not `libdb2.so`).

3. Define a DSN for the installed driver by editing the `/etc/odbc.ini` file, as in the following example:

```
[Postgre_DSN]
Description = Test
Driver = /usr/lib64/psqlodbc.so
Endpoint = MyDatabase
Servername = 12.3.45.678
Port = 5432
```

ODBC target data types

The following table shows the ODBC target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.



*ODBC does not support applying changes to binary data types in **Batch optimized apply** mode. For more information on **Batch optimized apply** mode, see [Change Processing Tuning \(page 1181\)](#).*

For information on how to view the data type that is mapped from the source, see the section for the source endpoint you are using.

For additional information about Qlik Replicate data types, see [Replicate data types \(page 81\)](#).

Supported data types

Qlik Replicate Data Types	ODBC Data Types
BOOLEAN	SQL_BIT
BYTES	SQL_VARBINARY
DATE	SQL_TYPE_DATE
TIME	SQL_TYPE_TIME
DATETIME	SQL_TYPE_TIMESTAMP
INT1	SQL_SMALLINT
INT2	SQL_SMALLINT
INT4	If the target endpoint supports precision and scale then: SQL_INTEGER Otherwise: SQL_VARCHAR
INT8	SQL_BIGINT
NUMERIC	SQL_NUMBER
REAL4	SQL_REAL
REAL8	SQL_DOUBLE
STRING	SQL_VARCHAR
UINT1	SQL_TINYINT
UINT2	SQL_SMALLINT
UINT4	SQL_INTEGER
UINT8	SQL_BIGINT

Qlik Replicate Data Types	ODBC Data Types
WSTRING	SQL_WVARCHAR
CLOB	CLOB
BLOB	BLOB



If the target endpoint does not support ODBC data types, data types are mapped to SQL_VARCHAR.

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add an ODBC endpoint to Qlik Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button.
2. In the **Name** field, type a name for your ODBC endpoint. This can be any name that will help to identify the endpoint being used.
3. In the **Description** field, type a description that helps to identify the ODBC endpoint. This is optional.
4. Select **Target** as the endpoint **role**.
5. Select **ODBC** as the endpoint **Type**.
6. Select one of the following:
 - **DSN**: Select this to connect to an ODBC-supported endpoint using a DSN. When you select DSN you must select the DSN you are using from the list.



When connecting to SQL/MP, you must use a connection string, which should include the name of the Replicate ARC Unicode ODBC driver. See [Connection String](#) for an example.

If the DSN you want to use is not included in the list, make sure that the endpoint client is installed on the computer with Qlik Replicate and that the DSN is defined. Note that the ODBC provider client must be 64-bit. For more information, see *Prerequisites* (page 398).



If you are using an ARC CDC Agent as the source in a Replicate task, you cannot select the DSN for the Qlik ODBC driver as the target. In this case, to use Qlik ODBC as a target, you must enter the connection string manually by selecting **Connection String** and following the directions for that option in this procedure.

- **Connection String**: Select this to connect to an ODBC-supported endpoint using a connection string then type a valid connection string in the field below. For information on how to create a connection string, see the documentation for the ODBC endpoint provider you are using.

Example of an SQL/MP Connection String:

10 Adding and managing target endpoints

```
Driver={Qlik Replicate ARC ODBC Driver 3.5 (Unicode)};BindUrl=attconnect://ais_
server_ip:ais_server_port/ais_workspace;DefTdpName=ais_target_datasource_
name;OneTdpMode=1;qptdpname=BINDURL1;queryProcessor/noThreads=true;}}
```

Note that if you specify a password in your connection string, it will be revealed as plain text in the task log files. It is therefore recommended to specify the password in the GUI **Password** field.



- You can use the **Advanced** tab to add specific properties and create a custom connect string. In this case, you do not need to enter information in this tab. For more information on using the **Advanced** tab, see *Setting advanced connection properties* (page 964).
- To determine if you are connected to the endpoint you want to use or if the connection information you entered is correct, click **Test Connection**. If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.
To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

7. Type the authentication information (**User Name, Password**) for the authorized user for the ODBC endpoint being used. For example, the IBM DB2 system administrator if you are using a IBM DB2 provider. If you do not know this information, see your ODBC Endpoint System Administrator.



- When you select **Connection String** be sure to include **User name/password** information in the connection string that you type in the box.
If you are using the **Advanced** tab to create a custom string, make sure to include the **User Name** and **Password** properties. For more information, see *Setting advanced connection properties* (page 964).
- This information is case sensitive.
- You can set custom properties in the **Advanced** tab. For more information, see *Setting advanced connection properties* (page 964).



Make sure that the ODBC endpoint user has the correct access privileges for the ODBC provider being used.

Setting advanced connection properties

In the **Advanced** tab, you can set the following properties:

- **Provider syntax:** Select the name of the provider syntax. Note that when replicating to an HP NonStop SQL/MP target, you must select **SQLMP (ARC)** as the provider type.
- **Load using CSV:** Select to load the data using a CSV file.

- **Max file size** (KB): Select or type the maximum size (in KB) of a CSV file before the file is moved into the load folder. The default value is 32000 KB.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.38 Using Oracle as a target

This section describes how to set up and use an Oracle database as a target endpoint in a replication task.



When replicating to an Oracle database with a full disk and/or partition where Oracle is trying to write archived redo log files, insert operations may fail. In such a case, no error will be shown and the task will not progress past the loading stage. To confirm that this is an Oracle Archiver error, stop and attempt to restart the task. The task will not start and an appropriate error should be shown.

In this section:

- *Supported Oracle database editions (page 967)*
- *Client prerequisites (page 967)*
- *Limitations (page 968)*
- *Security requirements (page 969)*
- *Oracle target data types (page 969)*
- *Setting general connection properties (page 971)*
- *Setting advanced connection properties (page 973)*

Supported Oracle database editions

Before you begin to work with an Oracle database as a source or target in Qlik Replicate, make sure that the Oracle database with the tables that are necessary for replication is available in your system. Qlik Replicate supports the following Oracle database editions:

- Oracle Enterprise Edition
- Oracle Standard Edition
- Oracle Standard Edition 2
- Oracle Express Edition
- Oracle Personal Edition

Client prerequisites

Before you can work with an Oracle endpoint, make sure the prerequisites listed in this section have been met.



Support for the XMLTYPE data type requires the full Oracle Client.

- On Windows systems, install Oracle Instant Client for Microsoft Windows (x64) Version 12.0 or later.
- On Linux systems, install Oracle Instant Client for Linux (x86-64) Version 12.0 or later.
In addition, if not already included in your system, you need to create a symbolic link in the `$oracle_home/lib` directory. This link should be called `libclntsh.so`, and should point to a specific version of this file.

For example, on an Oracle 12c client:

```
lrwxrwxrwx 1 oracle oracle 63 Oct 2 14:16 libclntsh.so ->
/u01/app/oracle/home/lib/libclntsh.so.12.1
```

Additionally, append the `LD_LIBRARY_PATH` environment variable to the Oracle `lib` directory by copying the driver location to the `site_arep_login.sh` file as follows:

```
echo "export LD_LIBRARY_PATH=\$LD_LIBRARY_PATH:/u01/app/oracle/home/lib/" > site_arep_
login.sh
```



When Replicate Server is installed on Red Hat 8.1, you must install Oracle Instant Client for Linux (x86-64) Version 12.

Connecting to an Oracle autonomous data warehouse

The following procedure should be performed when replicating to an Oracle Autonomous Data Warehouse.

To set up connectivity to Oracle Autonomous Data Warehouse (ADW), you need the Client Credentials (Wallet) ZIP file that you downloaded when you set up your Oracle Autonomous Data Warehouse environment.

To enable Replicate to connect to Oracle Autonomous Data Warehouse:

1. Extract the contents of the Client Credentials ZIP file to a folder of your choice.
This can be any folder, but for the purpose of this procedure, `C:\myapp\adw_credentials` will be the Windows folder containing the extracted files while `/home/adw_credentials` will be the Linux folder containing the extracted files.
2. In the folder containing the extracted files, edit the `sqlnet.ora` file, replacing `*/network/admin` with the path of the folder containing the extracted files:
Replicate on Windows:
`WALLET_LOCATION = (SOURCE = (METHOD = file) (METHOD_DATA = (DIRECTORY="C:\myapp\adw_credentials")))`
`SSL_SERVER_DN_MATCH=yes`
Replicate on Linux:
`WALLET_LOCATION = (SOURCE = (METHOD = file) (METHOD_DATA = (DIRECTORY="/home/adw_credentials")))`
`SSL_SERVER_DN_MATCH=yes`
3. Create the `TNS_ADMIN` environment variable and set it to the location of the credentials file.
Replicate on Windows:
Set the `TNS_ADMIN` to `C:\myapp\adw_credentials`
Replicate on Linux:
Run the following commands:
`export TNS_ADMIN=/home/adw_credentials`
`export PATH=$TNS_ADMIN:$PATH`
4. In the [Connection String](#) field in the endpoint's **General** tab, enter one of the names from the `tnsnames.ora` file.

For more information on setting up connectivity to Oracle ADW, see:

[Connect Autonomous Transaction Processing Using a Client Application](#)

Limitations

The following limitations apply:

- The **Use direct path full load** option does not support the following:
 - Tables with `INDEXTYPE CONTEXT`
Workaround: Use Array Load.
 - Bidirectional replication
 - Triggers
- Qlik Replicate cannot create a new schema on the Oracle database. To replicate to a new schema, the new schema name must already exist on the Oracle target. You can then specify the new schema name in the Task Settings' *Target metadata* (page 1166) and *Control tables* (page 1169) tabs as required.
- When working in [Batch optimized apply](#) mode, empty LOB columns might be replicated as NULL values.
- After Full Load completes, if the **Limit LOB size to** option is enabled and a table contains NCLOB data, the NCLOB size on the target will be truncated to half the size specified in the [Limit LOB size to](#) field. CDC will still truncate it according to the limit.

Security requirements

A user must have the following privileges granted in the Oracle database to use an Oracle target in a Qlik Replicate task:

- CREATE ANY TABLE
- CREATE ANY INDEXES
- ALTER ANY TABLE
- DROP ANY TABLE
- INSERT ANY TABLE
- UPDATE ANY TABLE
- DELETE ANY TABLE
- SELECT ANY TABLE
- SELECT all_indexes
- SELECT all_ind_columns
- SELECT all_constraints
- SELECT all_cons_columns
- LOCK ANY TABLE (Required for the **Use direct path full load** option in the **Advanced** tab).

Oracle target data types

The Oracle database for Qlik Replicate supports most Oracle data types. The following table shows the Oracle target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source database you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

Data types

Qlik Replicate Data Types	Oracle Data Types
BOOLEAN	NUMBER (1)
BYTES	RAW (length)
DATE	DATETIME
TIME	TIMESTAMP (0)
DATETIME	If scale <= 9, then TIMESTAMP (scale) Otherwise, VARCHAR2 (37)
INT1	NUMBER (3)
INT2	NUMBER (5)
INT4	NUMBER (10)

10 Adding and managing target endpoints

Qlik Replicate Data Types	Oracle Data Types
INT8	NUMBER (19)
NUMERIC	NUMBER (p,s)
REAL4	BINARY_FLOAT
REAL8	BINARY_DOUBLE
STRING	<p>With date indication: DATE</p> <p>With time indication: TIMESTAMP</p> <p>With timestamp indication: TIMESTAMP</p> <p>With timestamp_with_timezone indication: TIMESTAMP WITH TIMEZONE</p> <p>With timestamp_with_local_timezone indication: TIMESTAMP WITH LOCAL TIMEZONE</p> <p>With interval_year_to_month indication: INTERVAL YEAR TO MONTH</p> <p>with interval_day_to_second indication: INTERVAL DAY TO SECOND</p> <p>If Length > 4000: CLOB</p> <p>In all other cases: VARCHAR2 (Length)</p>
UINT1	NUMBER (3)
UINT2	NUMBER (5)
UINT4	NUMBER (10)
UINT8	NUMBER (19)
WSTRING	<p>NVARCHAR2 (length)</p> <p>Note that when length is greater than 2000, the column data type will be NCLOB.</p>
BLOB	<p>BLOB</p> <p>To use this data type with Qlik Replicate, you must enable the use of BLOBs for a specific task.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p>

Qlik Replicate Data Types	Oracle Data Types
CLOB	<p>CLOB</p> <p>To use this data type with Qlik Replicate, you must enable the use of CLOBs for a specific task.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p>
NCLOB	<p>NCLOB</p> <p>To use this data type with Qlik Replicate, you must enable the use of NCLOBs for a specific task.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p>
The XMLTYPE target data type is only relevant in Oracle-to-Oracle replication tasks. See the note below.	XMLTYPE



When the source database is Oracle, the source data types will be replicated "as is" to the Oracle target. For example, an XMLTYPE data type on the source will be created as an XMLTYPE data type on the target.

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.



The total number of columns per table supported in [Batch optimized apply](#) mode can be expressed using the following formula:

$$2 * \text{columns_in_original_table} + \text{columns_in_primary_key} \leq 999$$

So, for example, if the original tables has 25 columns and its Primary Key consists of 5 columns, then the total number of columns would be 55. If a table exceeds the supported number of columns, Replicate will apply all of the changes in one-by-one mode.

To add an Oracle target endpoint to Qlik Replicate:

1. In the Qlik Replicate console, click **Manage Endpoint Connections** to open the **Manage Endpoint Connections** dialog box. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints* (page 114).

2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Oracle database. This is optional.
4. Select **Target** as the database **role**.
5. Select **Oracle** as the database **Type**.
6. Type the Oracle **Connection String** for the Oracle database you want to work with.



- When replicating to an Oracle Autonomous Data Warehouse, you first need to perform the procedure described in [Connecting to an Oracle autonomous data warehouse \(page 967\)](#).
- This information is case sensitive.

You can type the connect string in any Oracle format, for example:

```
//host:port/service name
```

Where:

- **host:** This is the name or IP address for the computer with the Oracle database that you are using. For example, johnboy_w7 or 255.255.255.0.
- **port:** (optional) This is the TNS Listener Port number for the computer with the Oracle database that you are using. If you do not enter a port number the default Oracle TNS Listener port is used.
- **service name:** (optional) This is the service name for the computer with the Oracle database you are using. If you do not enter a service name the default service name is used.

You can also enter an Oracle Net keyword-value pair. For example:

```
"(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP) (HOST=d1sun242) (PORT=5521))  
(CONNECT_DATA=(SERVICE_NAME=bjava21)))"
```

7. Type the Oracle authentication information (**User Name, Password**) for the authorized user for this Oracle database. If you do not know this information, see your Oracle database Administrator (DBA). To ensure that the password provided to Replicate is never used by any other principal, Replicate can be configured to automatically replace the user-entered password with a strong random password. For more information, see [Configuring Replicate to Automatically Replace the User-Entered Password](#).



Make sure that the Oracle user entered in the Oracle Authentication section has the correct access privileges. For information on how to provide the required privileges, see [Security requirements \(page 969\)](#).



This information is case sensitive.



To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**.

If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

Setting advanced connection properties

You can set additional properties in the **Advanced** tab of the Oracle database connection settings.

You can set the following properties:

Use direct path full load: Select this to use the OCI direct path protocol for bulk loading Oracle tables. This is the default selection.



This option requires the user specified in the **General** tab to be granted the `LOCK ANY TABLE` permission.



Due to an issue with Oracle Direct Path, when this option is selected and **If target table already exists** is set to **Do nothing** in the Full Load Settings (page 1171), the following occurs:

- The first time the task runs, no error will be issued and rows with the same Primary Key may be added to the target table.
- The second time the task runs, the setting will take effect.
- Any subsequent times the task runs, an error will be generated.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.39 Using Pivotal Greenplum as a target

This section describes how to set up and use a Pivotal Greenplum database as a target in a replication task.

In this section:

- *An overview of the Pivotal Greenplum target (page 976)*
- *Qlik Replicate Pivotal Greenplum endpoint architecture overview (page 976)*
- *The Pivotal Greenplum Full Load process (page 978)*
- *Applying changes to the Pivotal Greenplum target (page 978)*
- *Prerequisites (page 978)*
- *Provide Pivotal Greenplum account access (page 979)*
- *Security requirements (page 980)*
- *Limitations (page 980)*
- *Pivotal Greenplum data types (page 980)*
- *Setting up the gpfdist program as a service (page 982)*
- *Using multiple gpfdist programs (page 983)*
- *Setting general connection properties (page 983)*
- *Setting advanced connection properties (page 985)*
- *Testing your Pivotal Greenplum Configuration and Environment (page 986)*

An overview of the Pivotal Greenplum target

The Qlik Replicate database for Pivotal Greenplum is a powerful operational data warehousing solution that manages Big Data analytics and challenges. Qlik Replicate uses Pivotal Greenplum's Scatter/Gather Streaming technology to help with data integration. This technology handles large amounts of data well.

The Qlik Replicate Pivotal Greenplum database makes it possible to load data from other heterogeneous data sources and maintain the most up to date information. This is done by capturing changes and streaming the changes to the Pivotal Greenplum data warehouse. This can be done with a very low impact on the source data.

The Qlik Replicate Pivotal Greenplum database provides full automation for:

- Schema generation and data type mapping
- Full load of source database tables
- Incremental load of changes made to source tables
- Application of DDL changes made to the source tables.
- Synchronization between full load and CDC processes.

Manual control is also available if needed.

The Qlik Replicate Pivotal Greenplum database integrates with the Pivotal Greenplum database in two ways:

- **Pivotal Greenplum ODBC API.** This is used for metadata management. The Pivotal Greenplum ODBC API lets Qlik Replicate test the database connection, get the table list and the table schema, build procedures that create external tables to process a file, and invoke the procedures that load the destination table or apply changes from the external table. During the schema generation, data types can be mapped, such as Pivotal Greenplum to Postgres. Primary keys and distribution clauses are generated based on the primary key.
- **Pivotal Greenplum Parallel File Distribution Server (gpfdist).** This utility is used with read-only external tables for fast, parallel data loading into a Pivotal Greenplum data warehouse. gpfdist uses maximum parallelism while reading from external tables. Qlik Replicate works closely with gpfdist to take advantage of its optimized fast, parallel loading facilities. Qlik Replicate uses the Pivotal Greenplum Parallel File Distribution Server to support both full load and incremental load activities.

See *Qlik Replicate Pivotal Greenplum endpoint architecture overview (page 976)* for a description of the system architecture used with the Pivotal Greenplum database.

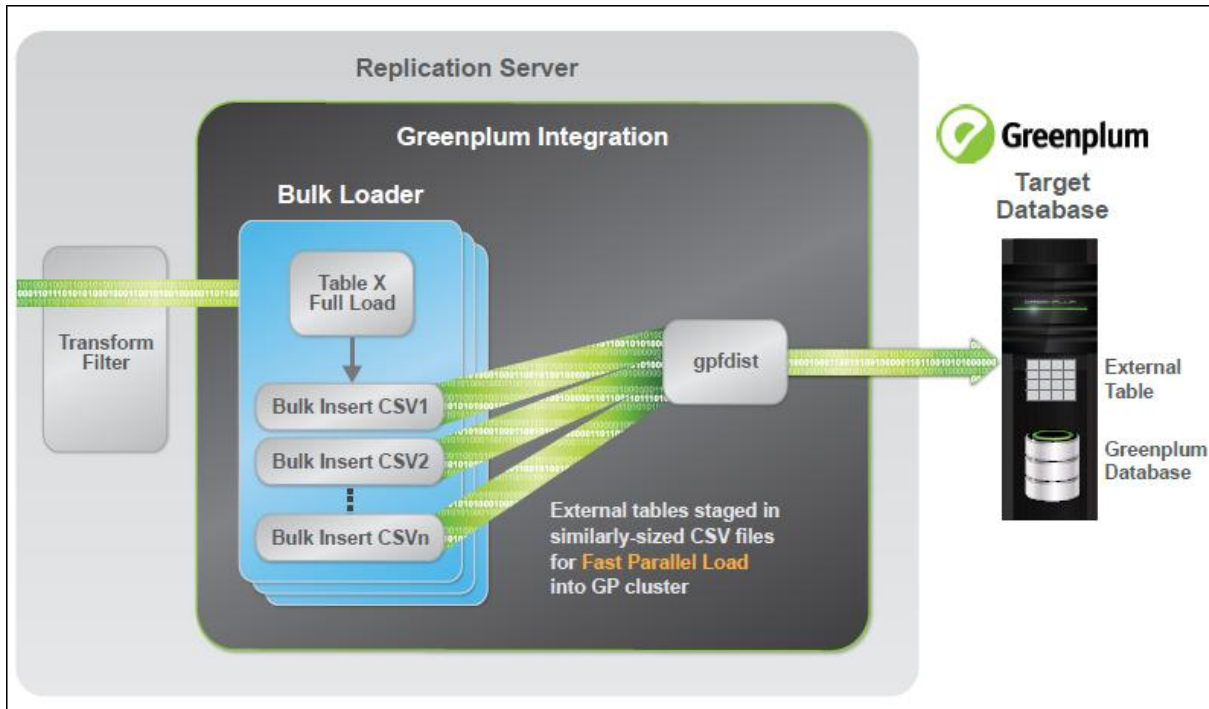
Qlik Replicate Pivotal Greenplum endpoint architecture overview

The following shows the Qlik Replicate Pivotal Greenplum endpoint system architecture for:

- *Full load (page 977)*
- *CDC (page 977)*

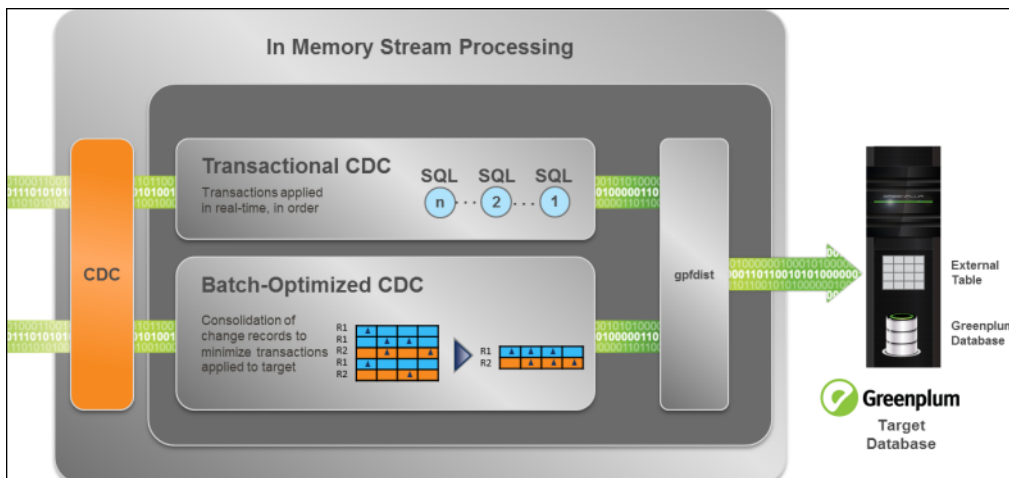
Full load

Full load is used to setup or refresh a data warehouse on a target by concurrently loading large amounts of data from source tables. High-speed data extraction is initiated from endpoints like Oracle or Microsoft SQL Server, then gpfdist and buffered load files are used for high-speed data loading into Pivotal Greenplum. The following shows the Pivotal Greenplum database architecture for full load.



CDC

For incremental load, Qlik Replicate uses log-based change data capture (CDC). During CDC replication, Qlik Replicate creates external Web tables or external tables to load SQL statements into the target Pivotal Greenplum database. The statements are then applied to the target tables. The following shows the Pivotal Greenplum database architecture for CDC.



The Pivotal Greenplum Full Load process

On the first run of a task the Pivotal Greenplum target writes the data being replicated to CSV files into a folder that is defined for the task. The CSV files are named sequentially, for example, loadNNNN, where NNNN is an incremental number starting from 0. The maximum file size of the CSV file is set by the user when configuring the Pivotal Greenplum database.

When the CSV file reaches its maximum size it is renamed and moved into a load folder. It is then read by the gpfdist utility, which executes an SQL statement that loads the data into the target table. Once the file loading is complete, the file is deleted.

Applying changes to the Pivotal Greenplum target

You can apply changes in one of two modes:

- *Transactional apply Mode (page 978)*
- *Batch-Optimized Apply Mode (page 978)*

For information on how to use the transactional apply mode, see *Apply Changes Settings (page 1174)*.

Transactional apply Mode

In this mode, the Pivotal Greenplum database writes all the change records to CSV files as DML statement. When a file is ready, the Pivotal Greenplum database creates an external Web table that uses the gpfdist server to read changes from the file and executes the DML statements in each row returned from the external Web table. When the changes are applied, the file is deleted.

Batch-Optimized Apply Mode

In this mode, the Pivotal Greenplum database writes net changes only to CSV files. When a file is ready, the Pivotal Greenplum database uses an external table that uses the gpfdist server to read the net changes from the file to a temporary table. The net changes are then applied to the target tables in the most efficient way. When the changes are applied, the file is deleted.

Prerequisites

You can use the Pivotal Greenplum database with Qlik Replicate on either a Windows or Linux computer. The following topic describe the prerequisites for working with Qlik Replicate and a Pivotal Greenplum database.

Replicate on Windows: Required software

You must install the following software on the computer where the Qlik Replicate Server is installed:

- greenplum-db-clients-6.7.1
- PROGRESS_DATADIRECT_CONNECT64_ODBC_7.1.6.HOTFIX



To prevent errors during replication tasks, make sure that the Pivotal Greenplum ODBC driver has a valid license.

Replicate on Linux: Required software

On the Linux computer, perform the following steps:

1. Open port 8080.
2. Install the following Greenplum software:
 - greenplum-db-clients-6.7.1
 - PROGRESS_DATADIRECT_CONNECT64_ODBC_7.1.6.HOTFIX
3. Update the `odbcinst.ini` file located in the **etc** directory, as in the following *example*:

```
[DataDirect 7.1 Greenplum Wire Protocol]
Driver=/opt/Progress/DataDirect/Connect64_for_ODBC_71/lib/ddgplm27.so
Setup=/opt/Progress/DataDirect/Connect64_for_ODBC_71/lib/ddgplm27.so
APILevel=0
ConnectFunctions=YYY
DriverODBCVer=3.52
FileUsage=0
HelpRootDirectory=/opt/Progress/DataDirect/Connect64_for_ODBC_71/help
SQLLevel=0
```

4. Update the `/opt/attunity/replicate/bin/site_arep_login.sh` file so that the Pivotal Greenplum environment scripts are executed when logging into the account where Qlik Replicate is run.

Example:

```
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/Progress/DataDirect/Connect64_for_ODBC_71/lib
export GPHOME_CLIENTS=/usr/local/greenplum-db-clients
```

5. Make sure you create a symbolic link from **greenplum-db-clients** to the Greenplum ODBC client.

Example:

```
greenplum-db-clients -> /usr/local/greenplum-db-clients-6.7.1 (make sure that the path starts with /usr/local).
```

6. Restart Replicate Server.

Configuration and environment

Qlik Replicate relies on the proper functioning of the Pivotal Greenplum's `gpfdist` program on the computer with Qlik Replicate (the local computer). `gpfdist` is a simple Web server program with special performance customization for concurrent access from Pivotal Greenplum database segment nodes to data files on the local computer.

Because `gpfdist` is a Web server program and because it needs to be accessible from the Pivotal Greenplum database segment nodes, there are some networking configuration settings that must be in place to allow for this access. This is documented in the *EMC Pivotal Greenplum database Administration Guide*.

For further information, see *Testing your Pivotal Greenplum Configuration and Environment (page 986)*.

Provide Pivotal Greenplum account access

The Qlik Replicate user who is working with the Pivotal Greenplum database must be registered as a user in the Pivotal Greenplum database. This is the user that is entered in the dialog box when *Setting general connection properties (page 983)*. You must grant Pivotal Greenplum account access to this user before

configuring the database in Qlik Replicate.



One of the Qlik Replicate computer network cards must be part of the Pivotal Greenplum database segments network to allow communication with the gpfdist program.

See *Testing your Pivotal Greenplum Configuration and Environment (page 986)* for additional information about connecting to and configuring a Pivotal Greenplum database to work with Qlik Replicate.

Security requirements

A user must have the following privileges granted in the Pivotal Greenplum database to use a Pivotal Greenplum target in a Qlik Replicate task:

- CREATE table/external table/Web external table
- TRUNCATE table
- ALTER table
- INSERT/UPDATE/DELETE

Limitations

Qlik Replicate cannot update columns that are part of the distribution key.

The Pivotal Greenplum target database has limited LOB support. You cannot use an unlimited LOB size for this database. For more information, see *Pivotal Greenplum data types (page 980)*.

Pivotal Greenplum data types

The Pivotal Greenplum database for Qlik Replicate supports most Pivotal Greenplum data types. The following table shows the Pivotal Greenplum target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source database you are using. For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

Data types

Qlik Replicate Data Types	Pivotal Greenplum Data Types
BOOLEAN	bool
BYTES	bytea
DATE	date
TIME	time (p)
DATETIME	timestamp (p)

Qlik Replicate Data Types	Pivotal Greenplum Data Types
INT1	int2
INT2	int2
INT4	int4
INT8	int8
NUMERIC	numeric (p,s)
REAL4	float4
REAL8	float8
STRING	varchar (n); n=data length
UINT1	int2
UINT2	int4
UINT4	int8
UINT8	numeric (20)
WSTRING	varchar (n); n=length
BLOB	<p>bytea</p> <p>To use this data type with Qlik Replicate, you must enable the use of BLOBs for a specific task.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p> <p>The Pivotal Greenplum target database has limited LOB support. You cannot use an unlimited LOB size for this database.</p> <p>For more information, see Task Settings/Metadata (page 1166).</p>
CLOB	<p>text</p> <p>To use this data type with Qlik Replicate, you must enable the use of CLOBs for a specific task.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p> <p>The Pivotal Greenplum target database has limited LOB support. You cannot use an unlimited LOB size for this database.</p> <p>For more information, see Task Settings/Metadata (page 1166).</p>

Qlik Replicate Data Types	Pivotal Greenplum Data Types
NCLOB	<p>text</p> <p>To use this data type with Qlik Replicate, you must enable the use of NCLOBs for a specific task.</p> <p>During CDC or during Full Load when the Allow unlimited LOB size option is enabled, LOB data types are supported only in tables with a primary key or unique index.</p> <p>The Pivotal Greenplum target database has limited LOB support. You cannot use an unlimited LOB size for this database.</p> <p>For more information, see Task Settings/Metadata (page 1166).</p>

Setting up the gpfdist program as a service

You can use the gpfdist program as a service when replicating data to Pivotal Greenplum endpoints using Qlik Replicate. In this way it is assured that a replication task that is running will not lose its connection to gpfdist. This solution is best used when more than one task is using the same gpfdist program.

Note that you can set up Pivotal Greenplum endpoints as described in *Setting general connection properties* (page 983) and test it without creating a service, but you should create the service before you begin to work with the Pivotal Greenplum endpoints.



If you want to use multiple gpfdist programs, then you should set them up on different ports. See *Using multiple gpfdist programs* (page 983).

To create use a gpfdist service:

1. From the command-line console on the computer where Qlik Replicate is installed, change the directory to the directory where Qlik Replicate is installed.
2. Type the following at the command-line prompt to create the service.

```
repctl service create <name of service (optional)> database=<name of Pivotal Greenplum database>
```



The Pivotal Greenplum database that you use in this command must be configured in Qlik Replicate. If you have defined more than one Pivotal Greenplum database, you can use any of the database names. All of the defined endpoints on the same Qlik Replicate instance are included in the defined service.

3. Start the service before you run any Pivotal Greenplum tasks by typing the following at the command line prompt:

```
sc start AttunityReplicateServer_<name of database you defined the service with>
```



If you chose to create a data folder in a separate location from where you installed Qlik Replicate, you must add the prefix `-d <path to data folder>` before any command line task described in this section. For example, to start the service you must type:
`-d <path to data folder> sc start AttunityReplicateServer_<name of database>`
Working with a data folder in a different location is recommended when working with large Pivotal Greenplum endpoints. For more information, see *Setting up Qlik Replicate on Windows (page 40)*.

To stop the service:

- Type the following at the command-line prompt:
`sc stop AttunityReplicateServer_<name of database you defined the service with>`

To delete the service:

- Type the following at the command-line prompt:
`sc delete AttunityReplicateServer_<name of database you defined the service with>`



A log file is available in the Qlik Replicate data folder and in the Pivotal Greenplum debug files, which is accessed through the Pivotal Greenplum database console.

Using multiple gpfdist programs

You can define each replication task that has a Pivotal Greenplum database as a target to use a separate gpfdist program. In this case each gpfdist should run on a different port. If you want to use the same gpfdist program for each task, you may want to use the gpfdist program as a service. For more information, see *Setting up the gpfdist program as a service (page 982)*.

To use gpfdist on multiple ports:

1. Install gpfdist programs on the computers you want to use them.
2. Make sure that you assign a different port number for the gpfdist program in each task. For more information, see *Setting general connection properties (page 983)*.

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a Pivotal Greenplum Target endpoint to Qlik Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoint Connections** dialog box.
2. In the **Manage database Connections** dialog box, click **New Endpoint Connection**.
3. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.

10 Adding and managing target endpoints

4. In the **Description** field, type a description that helps to identify the Pivotal Greenplum database. This is optional.
5. Select **Target** as the database **role**.
6. Select **Pivotal Greenplum** as the database **Type**.
7. Type the **Database host name**. This is the name of the computer with the Pivotal Greenplum instance you want to work with.



*Although the gpfdist program acts as a Webserver, it does not carry out security checks on any requests made to it. Therefore, when you define the path to the gpfdist program, it must be to a specific location so that no other data on the computer is accessed. You can use the **Advanced** tab to add specific properties and create a custom connect string. In this case, you do not need to enter information in this tab. For more information on using the **Advanced** tab, see *Setting advanced connection properties* (page 985).*

8. Type the Pivotal Greenplum database Port, where the Pivotal Greenplum instance you are working with is located. The default value is 5432.
9. Type the Pivotal Greenplum authentication information (**User Name, Password**) for the authorized user for this Pivotal Greenplum database. If you do not know this information, see your Pivotal Greenplum system manager.



- *If you are using the **Advanced** tab to create a custom string, make sure to include the **User Name** property. A **Password** can also be included but is not required. See *Setting advanced connection properties* (page 985) for more information.*
- *This information is case sensitive.*
- *If you want to set custom properties for this database, see *Setting advanced connection properties* (page 985).*
- *To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**. If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box. To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.*



*Make sure that the Pivotal Greenplum user entered in the Pivotal Greenplum Authentication section has the correct access privileges. For information on how to provide the required privileges, see *Security requirements* (page 980).*

10. Type the **Database name** or select one from the list of available databases. This is the name of the Pivotal Greenplum database where you are replicating the data to.
11. Type the **gpfdist hostname** for the server where the gpfdist program is installed.
12. Type the **gpfdist port** number where the gpfdist program is listening. The default value is 8080.

13. In the **Security** section:
 1. To enable SSL, select the **Use SSL** check box.
 2. In the **CA Path**, specify the folder containing the certificate required to execute gpfdist.
 3. In the **Public key file** field, specify the full path of the public key file (i.e. including the file name). The file can reside in the same folder as the CA certificate.
 4. In the **Private key file** field, specify the full path of the private key file (i.e. including the file name). The file can reside in the same folder as the CA certificate.

Setting advanced connection properties

In the **Advanced** tab, you can set the following parameters:

- **gpfdist max row length**: Select or type the maximum length of a row (number of characters) in the CSV file that is sent to the gpfdist program. This is the maximum row length read by gpfdist. The default value is 32,768. The larger the size of the rows the more resources the gpfdist program uses.
- **Create tables in tablespace**: Type the name of the tablespace where you want create the target tables. This is optional. Note that the tablespace that you enter must exist in the Pivotal Greenplum database.
- **Max file size** (KB): Select or type the maximum size (in KB) of a CSV file before the file is moved into the load folder. The default value is 32000 KB.
- **Write buffer size**: (KB): Select or type the maximum amount of memory (in Kilobytes) used to store the data before moving it to the load folder. The default value is 1001.
- **ODBC driver**: Type the name of the ODBC driver you are using to connect to the Pivotal Greenplum database you are working with. The default value is **DataDirect 7.0 Pivotal Greenplum Wire Protocol**.
- **Additional ODBC connection properties**: Type any additional ODBC connection properties if required
- **Use externally managed gpfdist**: Select this check box to use an external gpfdist with this Pivotal Greenplum database.
- **Storage folder**: Type the name and location (enter full path) of the folder that holds the CSV files for loading. This is available only if you are using an externally managed gpfdist.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

Testing your Pivotal Greenplum Configuration and Environment

Qlik Replicate relies on the proper functioning of Pivotal Greenplum's gpfdist program on the computer with Qlik Replicate (the local computer). gpfdist is a simple Web server program with special performance customizing for concurrent access from Pivotal Greenplum database segment nodes to data files on the local computer.

Because gpfdist is a Web server program and because it needs to be accessible from the Pivotal Greenplum database segment nodes, there are some networking configuration settings that must be in place to allow for this access. This is documented in the *EMC Greenplum database Administration Guide*.

The following sections provide a simple test that verifies the proper configuration of the Pivotal Greenplum database and the local software installation. You can run this test before installing any Qlik software on your local computer. This test must be completed successfully to ensure that Qlik Replicate can work with a Pivotal Greenplum database.

- *Collect Connection Information (page 986)*
- *Create a Test Input File (page 987)*
- *Create an SQL Script File (page 987)*
- *Start gpfdist (page 987)*
- *Run the SQL Script (page 987)*

Collect Connection Information

The following is the information required for this test. You should write down the information for your system to use later in this test.

- `<<Pivotal Greenplum-host>>`: The Pivotal Greenplum database host name (master).
- `<<Pivotal Greenplum-port>>`: The Pivotal Greenplum database port number (master). In many cases the port number is 5432
- `<<Pivotal Greenplum-user>>`: The username that is used to connect to the Pivotal Greenplum database. This user must have permission to create an external table.
- `<<Pivotal Greenplum-password>>`: The password of the selected Pivotal Greenplum user.
- `<<Pivotal Greenplum-database>>`: The Pivotal Greenplum database name where the test is created.

- `<<Pivotal Greenplum-host>>`: The local computer name as seen from the Pivotal Greenplum segment nodes.
- `<<gpfdist-port>>`: The port number where the gpfdist program is listening.

Create a Test Input File

Create a text file called `greenplum_test123.txt` with the following content:

```
Scott,Greenplum
```

```
Tiger,Woods
```

Create an SQL Script File

Create a text file called `greenplum_test123.sql` with the following content, replacing the `<<.fgf.>>` tokens with the information for your system that you wrote down in step 1:

```
CREATE EXTERNAL TABLE greenplum_test123 ( name text, descr text )
LOCATION ('gpfdist://<<gpfdist-host>>:<<gpfdist-port>>/greenplum_test123.txt')
FORMAT 'TEXT' (DELIMITER ',');
SELECT * FROM greenplum_test123;
DROP EXTERNAL TABLE greenplum_test123;
```

Start gpfdist

Open a command shell and change the current directory to the directory where you created the Test Input File and the SQL Script File. Then start the gpfdist program with the following command:

```
$ gpfdist -v -p <<gpfdist-port>> -d .
```

Run the SQL Script

Run the SQL script that you created with the following command:

```
$ psql -d <<Pivotal Greenplum-database>> -h <<Pivotal Greenplum-host>> -p <<Pivotal Greenplum-
port>> -U <<Pivotal Greenplum-user>> -f greenplum_test123.sql
```

If the script runs successfully, the following is displayed:

```
CREATE EXTERNAL TABLE
name | descr
-----+-----
Scott | Pivotal Greenplum
Tiger | woods
(2 rows)
DROP EXTERNAL TABLE
```

If the script is not successful, you will get an output that is similar to the following example:

```
CREATE EXTERNAL TABLE
psql:greenplum_test123.sql:4: ERROR: connection with gpfdist failed for
gpfdist://atturep1:18080/greenplum_test123.txt. effective url: http://192.168.165.12:
18080/greenplum_test123.txt. error code = 110 (Connection timed out) (seg0 slice1
greenplum421.acme.local:40000 pid=31364)
DROP EXTERNAL TABLE
```

In the example above, the problem was that a firewall on the local computer prevented the Pivotal Greenplum database segment from reaching the local gpfdist instance.

Any error that occurs must be resolved before using Qlik Replicate with EMC Pivotal Greenplum. Once you resolve the error, you should run this test again to ensure that you can work with the Pivotal Greenplum database.

For information about what to do if this test fails, see *Troubleshooting gpfdist issues* (page 988).

Troubleshooting gpfdist issues

If the test described in *Testing your Pivotal Greenplum Configuration and Environment* (page 986) fails, you should carry out the following checks:

- *Did gpfdist start on the correct port or protocol?* (page 988)
- *Can Pivotal Greenplum reach gpfdist?* (page 988)

Did gpfdist start on the correct port or protocol?

To check whether gpfdist is listening on the correct port or protocol, enter the following command:

For Windows:

```
$ netstat -a -n | find "<<gpfdist-port>>"
```

For Linux:

```
$ netstat -a -n | grep "<<gpfdist-port>>"
```

The following is the output that you should get if gpfdist started on port 8080 on Windows:

```
$ netstat -a -n | find "8080"  
TCP 0.0.0.0:8080 0.0.0.0:0 LISTENING  
TCP [::]:8080 [::]:0 LISTENING
```

This indicates that gpfdist is listening on any network interface on both IPv4 (0.0.0.0:8080) and IPv6 ([::]:8080).

If only the IPv6 line is shown in most cases there is a local networking configuration problem. Pivotal Greenplum's recommendation in this case is to disable IPv6 on the local computer (see the Microsoft knowledge base article on how to carry this out at <http://support.microsoft.com/kb/929852>).

Can Pivotal Greenplum reach gpfdist?

For gpfdist to work, all Pivotal Greenplum database segments must be able to communicate with the local machine (the **ETL machine** as described by Pivotal Greenplum) using the HTTP protocol.

While gpfdist is running, run the following command from each one of the Pivotal Greenplum database segment nodes to ensure that all segments can access gpfdist:

```
$ wget http?//<<Pivotal Greenplum-host>>?<<Pivotal Greenplum-port>>/gpfdist/status
```

This should return a status page with the following or similar content:

```
read_bytes 0  
total_bytes 0  
total_sessions 0
```

When carrying out the network checks from the Pivotal Greenplum internal master node, there is usually a file called `seg_host` that contains a list of the Pivotal Greenplum database segment node names. If this file exists, you can check access from all segment nodes using a single Pivotal Greenplum `gpssh` command:

```
$ gpssh -f seg_host
```


10 Adding and managing target endpoints

=> `wget http://<<Pivotal Greenplum-host>>:<<Pivotal Greenplum-port>>/gpfdist/status`

If this check fails, a network or system manager must change the network or system configuration so that the check succeeds.

10.40 Using a PostgreSQL-Based database as a target

This section describes how to set up and use a PostgreSQL-based target endpoint in a replication task.

You need to configure a PostgreSQL-based endpoint when replicating to any of the following databases:

- PostgreSQL
- Amazon Aurora (PostgreSQL)
- Amazon RDS for PostgreSQL
- Azure Database for PostgreSQL - Flexible Server


In this section:

- *Prerequisites (page 991)*
- *Limitations (page 992)*
- *Security requirements (page 992)*
- *PostgreSQL Database target data types (page 992)*
- *Setting general connection properties (page 993)*
- *Setting advanced connection properties (page 995)*

Prerequisites

This topic describes the client prerequisites when replicating to a PostgreSQL target.

Qlik Replicate Server for Windows

- **The PostgreSQL ODBC Driver:** PostgreSQL ODBC Driver [psqlodbc_13_01_0000-x64](#) or above must be installed on the Qlik Replicate machine.
- **PgAdmin:**
 1. Download and install  [PgAdmin 4 \(Windows\)](#) on the Qlik Replicate machine.
 2. Add the "psql.exe" path to the system PATH.
- Microsoft Visual Studio 2013 Redistributable (x86) - *not* (x64) - must be installed on the Qlik Replicate machine.

Qlik Replicate Server for Linux

Download PostgreSQL 13 ODBC driver, and client programs and libraries.

- **Example PostgreSQL ODBC file name:** postgresql13-odbc-13.02.0000-1PGDG.rhel8.x86_64.rpm
- **PostgreSQL client programs and libraries file name:** postgresql13-13.8-1PGDG.rhel<version>.x86_64.rpm

Then, open a shell prompt and do the following:

1. Stop the Replicate service:
`./areplicate stop`
2. Optionally, confirm that the service has stopped:
`./areplicate status`
3. Install PostgreSQL 13 client programs and libraries.
4. Install the ODBC driver for PostgreSQL 13.
5. Change the working directory to `<Replicate-Install-Dir>/bin`.
6. Copy the driver location to the `site_arep_login.sh` file:
`echo "export LD_LIBRARY_PATH=\$LD_LIBRARY_PATH:/usr/lib:/usr/lib64:/usr/pgsql-13/lib" >> site_arep_login.sh`
This will add the driver to "LD_LIBRARY_PATH" and update the driver location in the **site_arep_login.sh** file.
7. Optionally, confirm that the driver location was copied:
`cat site_arep_login.sh`
8. Makes sure that the `/etc/odbcinst.ini` file contains an entry for PostgreSQL, as in the following example:

```
[PostgreSQL Unicode(x64)]
Description = PostgreSQL ODBC driver
Driver = /usr/pgsql-13/lib/psqlodbc.so
Setup = /usr/pgsql-13/lib/psqlodbcw.so
UsageCount = 1
```
9. Start the Replicate service:
`./areplicate start`
10. Optionally, confirm that the service has started:
`./areplicate status`

Limitations

- UPDATE and DELETE operations on tables without a Primary Key/Unique Index that contain duplicate records is not supported. Any changes to these tables will affect all the records in the target.
- Tables with a LOB column in the key are not supported in Batch Optimized Apply mode. When a VARBINARY column is used as a source table key, a BYTEA column will be created in the target. This causes unpredictable behavior in Batch Optimized Apply mode. As a workaround, we suggest using the SQLite HEX function to convert VARBINARY to VARCHAR.

Security requirements

The user specified in the **General** tab when *Setting general connection properties (page 993)* must be a registered user in the PostgreSQL database.

In addition, the following privileges must be granted:

- Can login
- Create databases



If the target schema already exists in the database, the "Create databases" privilege is not required.

PostgreSQL Database target data types

The PostgreSQL endpoint for Qlik Replicate supports most PostgreSQL database data types. The following table shows the PostgreSQL database target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source database you are using. For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

Data types

Qlik Replicate Data Types	PostgreSQL database Data Types
BOOL	BOOL
BYTES	BYTEA
DATE	DATE
TIME	TIME
DATETIME	If scale is => 0 and =< 6, then: TIMESTAMP If scale is => 7 and =< 12, then: VARCHAR (37)
INT1	SMALLINT

Qlik Replicate Data Types	PostgreSQL database Data Types
INT2	SMALLINT
INT4	INTEGER
INT8	BIGINT
NUMERIC	DECIMAL (P, S)
REAL4	FLOAT4
REAL8	FLOAT8
STRING	If length is 1 - 21845, then: VARCHAR (Length in Bytes = The STRING value multiplied by three) If length is 21846 - 2147483647, then: VARCHAR (65535)
UINT1	SMALLINT
UINT2	INTEGER
UINT4	BIGINT
UINT8	BIGINT
WSTRING	If length is 1 - 21845, then: VARCHAR (Length in Bytes = The WSTRING value multiplied by three) If length is 21846 - 2147483647, then: VARCHAR (65535)
BLOB	BYTEA
NCLOB	TEXT
CLOB	TEXT

Data types when replicating from a PostgreSQL source

When replicating from a PostgreSQL source, the target table will be created with the same data types for all columns, apart from columns with user-defined or PostGIS data types. In such cases, the data type will be created as "character varying" in the target.

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a PostgreSQL target endpoint to Qlik Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the PostgreSQL database. This is optional.
4. Select **Target** as the database **role**.
5. From the **Type** drop-down list, select **PostgreSQL** for PostgreSQL, Amazon Aurora PostgreSQL, and Amazon RDS for PostgreSQL.
6. Type the **Server** name. This is the name or IP address or host name of the computer with the PostgreSQL database that you want to access.
7. Optionally, change the default port (5432).
8. Enter the PostgreSQL database authentication information (**User name**, **Password**) of an authorized PostgreSQL user. If you do not know this information, see your PostgreSQL database system manager.



Consider the following:

- *This information is case sensitive.*
- *To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**. If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box. To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.*



Make sure that the specified PostgreSQL database user has the correct access privileges.

9. Type the **Database name** or select one from the list of available endpoints. This is the name of the PostgreSQL database to which you are replicating data.

SSL authentication options

In the **Security** section, you can configure Replicate to connect to the PostgreSQL database using SSL.



The specified certificates are relevant only for bulk insert operations.

- **SSL Mode:** Select one of the following:
 - **Disable** - Connect with a surname and password only.
 - **Allow** - Establish an encrypted connection if requested by the server.
 - **Prefer** - Establishes an encrypted connection if the server supports encrypted connections, falling back to an unencrypted connection if an encrypted connection cannot be established.

- **Require** - Establishes an encrypted connection if the server supports encrypted connections. The connection attempt fails if an encrypted connection cannot be established.
- **Verify CA** - Similar to **Required**, but also verifies the server Certificate Authority (CA) certificate against the configured CA certificates. The connection attempt fails if no valid matching CA certificates are found.
- **Verify Full** - Similar to **Verify CA**, but also performs host name identity verification by checking the host name the client (i.e. Replicate) uses for connecting to the server against the identity in the certificate that the server sends to the client. The client checks whether the host name that it uses for connecting matches the Common Name value in the server certificate. The connection fails if there is a mismatch.
- **Client certificate path** - The path to the client certificate requested by the server.
- **Secret key path** - The path to the client private key file in PEM format.
- **CA path** - The path to the Certificate Authority (CA) that issued the client certificate file in PEM format.
- **CRL path** - The path to the CRL certificate. This file contains certificates revoked by certificate authorities. If the server certificate appears in this list, the connection will fail.
- **SSL compression** - Select this option to compress the data before it is encrypted.

Setting advanced connection properties

In the **Advanced** tab, you can set the following properties:

You can set the following properties:

- **Max file size (KB)**: Select or type the maximum size (in KB) of a CSV file before the file is loaded into the PostgreSQL target database. The default value is 32000 KB.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.41 Using SAP HANA as a target

This section describes how to set up and use a SAP HANA database as a target database in a replication task.

In this section:

- *Prerequisites (page 998)*
- *Limitations and considerations (page 998)*
- *Permissions (page 998)*
- *Supported data types (page 999)*
- *Setting general connection properties (page 999)*
- *Setting advanced connection properties (page 1000)*

Prerequisites

Windows

Install the SAP HANA ODBC 64-bit Driver 2.x for Windows on the Replicate Server machine. The driver name is HDBODBC.

Linux

Download the SAP HANA ODBC 64-bit Driver 2.x from the [SAP Software Download Center](#). You must have an account with SAP to download the software.

Example driver name: HDBODBC

Then, open a shell prompt and do the following:

1. Stop the Replicate service:
`./areplicate stop`
2. Optionally, confirm that the service has stopped:
`./areplicate status`
3. Install the SAP HANA ODBC 64-bit Driver 2.x for Linux on the Replicate machine.
4. Add the following section to the `odbcinst.ini` file located in directory `/etc`:
[HDBODBC]
Description=64-bit HANA ODBC Driver
Driver=/opt/sap/hdbclient/libodbcHDB.so
fileUsage=1
5. Start the Replicate service:
`./areplicate stop`
6. Optionally, confirm that the service has started:
`./areplicate status`

Limitations and considerations

When replicating to a SAP HANA target, the following limitations apply:

- INSERTs and UPDATEs of the BLOB data type are not supported in [Batch optimized apply](#) Change Processing mode. Use "Transactional apply" mode instead.
- Replicating data with 4-byte emoji characters is not supported.

Permissions

The user specified in the [SAP HANA endpoint settings](#) must be granted the following permissions:

- CREATE TABLES
- ALTER
- SELECT
- INSERT
- DELETE
- DROP

Supported data types

The Qlik Replicate SAP HANA target endpoint supports most SAP HANA data types. The following table shows the SAP HANA target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source database you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

Data types

Qlik Replicate Data Types	SAP HANA Data Types
BOOL	BOOLEAN
BYTES	VARBINARY (Length)
DATE	DATE
TIME	TIME
DATETIME	TIMESTAMP
INT1	TINYINT
INT2	SMALLINT
INT4	INTEGER
INT8	BIGINT
NUMERIC	DECIMAL (p,s)
REAL4	REAL
REAL8	DOUBLE
STRING	VARCHAR (Length)
UINT1	TINYINT
UINT2	SMALLINT
UINT4	INTEGER
UINT8	BIGINT
WSTRING	NVARCHAR (Length)
BLOB	BLOB
NCLOB	NCLOB
CLOB	CLOB

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a SAP HANA target endpoint to Qlik Replicate:

1. In the Qlik Replicate console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, optionally enter a description that helps to identify the SAP HANA database.
4. Select **Target** as the database **role**.
You can do this step before any of the other steps if you want, however before you can continue with the next step in this process, you must select the database **role**.
5. Select **SAP HANA** as the database **Type**.
6. In the **Server** field, specify the IP address or host name of the SAP HANA database server.
To connect to a High Availability Cluster, specify all of the cluster nodes and port numbers in the **Server** field. The nodes should be separated by a comma.
Example:
12.12.1.123:3033,12.12.1.124:3034
7. In the **Instance** field, enter the instance number of the target SAP HANA database.
8. Select either **Single tenant** or **Multi tenant** according to your **Database architecture**. If you selected **Multi tenant**, enter the target **Database name**.



The **Multi tenant** option is supported from Replicate May 2022 Service Release 02 only.

9. Enter the **Username** and **Password** required to access the SAP HANA database. If you do not know this information, see your SAP HANA database administrator (DBA).



This information is case sensitive.

10. Click **Test Connection** to verify that the specified settings are correct.

Setting advanced connection properties

In the **Advanced** tab, you can set internal parameters and view a summary of your settings for the SAP HANA target endpoint.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.42 Using SAP Sybase ASE as a target

This section describes how to set up and use a SAP Sybase ASE database as the target endpoint in a replication task.

In this section:

- *Prerequisites (page 1003)*
- *Limitations and considerations (page 1003)*
- *Security requirements (page 1003)*
- *SAP Sybase ASE database target data types (page 1003)*
- *Non-supported data types (page 1005)*
- *Setting general connection properties (page 1005)*
- *Setting advanced connection properties (page 1006)*

Prerequisites

This topic lists the prerequisites for using the SAP Sybase ASE target endpoint in a replication task.

- Install SAP Adaptive Server Enterprise ODBC Driver 16.0 (64-bit) on the Qlik Replicate machine.
- When replicating to SAP Sybase ASE 15.7 installed on a Windows machine configured with a non-Latin language (e.g. Chinese), Sybase 15.7 SP121 must be installed on the SAP Sybase ASE machine.
- When the Apply Changes replication option is enabled, either on its own or together with the Full Load replication option, you must specify the target schema in the task settings' *Target metadata* (page 1166) tab.

Limitations and considerations

The following limitations apply:

- Only one Qlik Replicate task can be run per SAP Sybase ASE database.
- Zero values located at the end of binary data type strings are truncated when replicated to the target database. For example, 0x00 in the source table will become 0x0001 in the target table.
- Qlik Replicate creates the target table with columns that do not allow NULL values, if the database default is not to allow NULL values. Consequently, if a Full Load or CDC replication task contains empty values, errors will occur.
To prevent this from happening:
 1. Right-click the database name and select **Properties** from the context menu.
 2. In the **Options** tab, select **Allow nulls by default** and then click **OK**.
- Replicating data with 4-byte emoji characters is not supported.

Security requirements

You must provide SAP Sybase ASE account access to the Qlik Replicate user. This user must have read/write privileges in the SAP Sybase ASE database.

SAP Sybase ASE database target data types

The following table shows the SAP Sybase ASE database target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.



SAP Sybase ASE does not support applying changes to binary data types in **Batch optimized apply** mode. For more information on **Batch optimized apply** mode, see *Change Processing Tuning* (page 1181).

For information on how to view the data type that is mapped from the source, see the section for the source endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).

10 Adding and managing target endpoints

Data types

Qlik Replicate Data Types	SAP Sybase ASE Data Types
BOOLEAN	When the BIT is nullable, then: TINYINT When the BIT is not NULL, then: BIT
BYTES	VARBINARY (Length)
DATE	DATE
TIME	TIME
DATETIME	If scale is $\Rightarrow 0$ and ≤ 6 , then: BIGDATETIME If scale is $\Rightarrow 7$ and ≤ 12 , then: VARCHAR (37)
INT1	TINYINT
INT2	SMALLINT
INT4	INTEGER
INT8	BIGINT
NUMERIC	NUMERIC (p,s)
REAL4	REAL
REAL8	DOUBLE PRECISION
STRING	VARCHAR (Length)
UINT1	TINYINT
UINT2	UNSIGNED SMALLINT
UINT4	UNSIGNED INTEGER
UINT8	UNSIGNED BIGINT
WSTRING	VARCHAR (Length)
BLOB	IMAGE
CLOB	UNITEXT
NCLOB	TEXT

Non-supported data types

Target SAP Sybase ASE tables with columns of the following SAP Sybase ASE data types cannot be replicated. Replicated columns with these data types will show as null.

- UDT

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a SAP Sybase ASE target endpoint to Qlik Replicate:

1. In the Qlik Replicate Console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the SAP Sybase ASE database. This is optional.
4. Select **Target** as the database **role**.
5. Select **SAP Sybase ASE** as the database **Type**.
6. In the **Server Name** field, enter the host name or IP address of the computer on which the SAP Sybase ASE database is installed.



Consider the following:

- *This information is case sensitive.*
- *You can use the **Advanced** tab to add specific properties and create a custom connect string. In this case, you do not need to enter information in this tab. For more information on using the **Advanced** tab, see *Setting advanced connection properties (page 1006)*.*
- *To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**. If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box. To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.*

7. Optionally, change the default port (5000).
8. Type the SAP Sybase ASE authentication information (**User Name**, **Password**) for the authorized user for this SAP Sybase ASE database. If you do not know this information, see your SAP Sybase ASE database Administrator (DBA).



Consider the following:

- This information is case sensitive.
- This information is required. If you are using the **Advanced** tab to create a custom string, make sure to include the **User Name** and **Password** properties. See *Setting advanced connection properties (page 1006)* for more information.
- If you want to set custom properties for this database, see *Setting advanced connection properties (page 1006)*.



Make sure that the SAP Sybase ASE user entered in the SAP Sybase ASE Authentication section has the correct access privileges. For information on how to provide the required privileges, see *Security requirements (page 1003)*.

9. In the **Database name** field, enter the SAP Sybase ASE database name.

Setting advanced connection properties

In the **Advanced** tab, you can set the following properties:

- **Additional ODBC connection properties:** Specify any additional ODBC connection parameters that you want to use.



If the user name or password specified in the **General** tab contains non-Latin characters (e.g. Chinese), the following property is required:

```
charset=gb18030
```

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

10 Adding and managing target endpoints

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.43 Using SAP Sybase IQ as a target

This section describes how to set up and use a SAP Sybase IQ database as a target database in a replication task.

In this section:

- *Prerequisites (page 1009)*
- *Limitations and considerations (page 1009)*
- *Security requirements (page 1009)*
- *SAP Sybase IQ target data types (page 1009)*
- *Setting general connection properties (page 1010)*
- *Setting advanced connection properties (page 1011)*

Prerequisites

Make sure the following prerequisites have been met:

- Qlik Replicate is installed on any Windows computer in your network.
- A Sybase account with the required access privileges exists.
- SAP Sybase IQ 64-bit ODBC client installed on the computer where Qlik Replicate is located.

Limitations and considerations

The following limitations apply:

- Full LOB mode is not supported.
- Replication of LOBs during Change Processing is not supported in Bulk Apply mode (LOB values are replicated to NULL).
- Replicating data with 4-byte emoji characters is not supported.

Security requirements

The user specified in the [SAP Sybase IQ endpoint settings](#) must have DBA permissions in the SAP Sybase IQ database.

SAP Sybase IQ target data types

The following table shows the Sybase target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.



SAP Sybase IQ does not support applying changes to binary data types in **Batch optimized apply** mode. For more information on **Batch optimized apply** mode, see *Change Processing Tuning* (page 1181).

For information on how to view the data type that is mapped from the source, see the section for the source database you are using.

For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).

Supported data types

Qlik Replicate Data Types	SAP Sybase IQ Data Types
BOOLEAN	TINYINT
BYTES	VARBINARY (Length)
DATE	DATE
TIME	TIME

Qlik Replicate Data Types	SAP Sybase IQ Data Types
DATETIME	If scale is => 0 and =< 6, then: TIMESTAMP If scale is => 7 and =< 12, then: VARCHAR (37)
INT1	TINYINT
INT2	SMALLINT
INT4	INTEGER
INT8	BIGINT
NUMERIC	NUMERIC (p,s)
REAL4	REAL
REAL8	DOUBLE
STRING	VARCHAR (Length)
UINT1	TINYINT
UINT2	SMALLINT
UINT4	INTEGER
UINT8	BIGINT
WSTRING	VARCHAR (Length)
BLOB	BLOB
CLOB	CLOB
NCLOB	CLOB



The SAP Sybase IQ database requires a special license to support LOBs.

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.



- *Sybase can also be used as a source database. For information on using Sybase as a source, see [Using SAP Sybase ASE as a target \(page 1002\)](#).*
- *You can also use Sybase files as a source or target. For more information, see [Using the Qlik Replicate file channel \(page 1061\)](#).*

To add a SAP Sybase IQ database to Qlik Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Sybase database. This is optional.
4. Select **Target** as the database **role**.
5. Select **SAP Sybase IQ** as the database **Type**.
6. In the **Host** field, enter the hostname or IP address of the computer on which SAP Sybase IQ is installed.
7. In the **Server** field, enter the name of the Sybase server.
8. Optionally, change the default port (7878).
9. Type the Sybase authentication information (**User Name**, **Password**) for the authorized user for this Sybase database. If you do not know this information, see your Sybase database Administrator (DBA).



- *This information is required. If you are using the **Advanced** tab to create a custom string, make sure to include the **User Name** and **Password** properties. See *Setting advanced connection properties (page 1011)* for more information.*
- *This information is case sensitive.*
- *If you want to set custom properties for this database, see *Setting advanced connection properties (page 1011)*.*



*Make sure that the Sybase user entered in the Sybase Authentication section has the correct access privileges. For information on how to provide the required privileges, see *Security requirements (page 1009)*.*

10. In the **Database** field, enter the name of the SAP Sybase IQ database.

Setting advanced connection properties

In the **Advanced** tab, you can set the following parameters:

- **Max file size:** Select or type the maximum size (in KB) of a CSV file before the file is loaded into the Sybase database. The default value is 32000 KB.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.

4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.44 Using SingleStore as a target

This section describes how to set up and use a SingleStore target endpoint in a replication task.

In this section:

- *Prerequisites (page 1014)*
- *Limitations and considerations (page 1014)*
- *Permissions (page 1015)*
- *Supported data types (page 1015)*
- *Setting general connection properties (page 1018)*
- *Setting advanced connection properties (page 1019)*

Prerequisites

Before using SingleStore as an endpoint in a Qlik Replicate task, make sure that the prerequisites outlined below have been met.

Driver setup

Installing the driver on Windows

Install MySQL ODBC Unicode Driver 64-bit version 8.0.32 on the Replicate machine.

Installing the driver on Linux

1. Stop the Replicate service:
`./areplicate stop`
2. Optionally, confirm that the service has stopped:
`./areplicate status`
3. Install MySQL ODBC Unicode Driver 64-bit version 8.0.32 on the Replicate machine.
4. Change the working directory to `<Replicate-Install-Dir>/bin`.
5. Copy the driver location to the `site_arep_login.sh` file:
`echo "export LD_LIBRARY_PATH=\$LD_LIBRARY_PATH:/usr/lib64" >> site_arep_login.sh`
This will add the driver to "LD_LIBRARY_PATH" and update the driver location in the **site_arep_login.sh** file.
6. Optionally, confirm that the driver location was copied:
`cat site_arep_login.sh`
7. Make sure that the `/etc/odbcinst.ini` file contains an entry for MySQL, as in the following example:

```
[MySQL ODBC 8.0 Unicode Driver]
Driver = /usr/lib64/libmyodbc8w.so
UsageCount = 1
```
8. Start the Replicate service:
`./areplicate start`
9. Optionally confirm that the service has started:
`./areplicate status`

Limitations and considerations

When replicating to a SingleStore target, the following limitations apply:

- Due to the way SingleStore operates, when loading data to a SingleStore target during a Full Load task, duplicate key errors will not be reported to the logs.
- Updating the Primary Key is not supported as the Primary Key columns in SingleStore also serve as the shard key.
- When updating a column's value to its existing value, a zero rows affected is returned from SingleStore (unlike Oracle and Microsoft SQL Server, for example, that perform an update of one row). This generates an entry in the `attrep_apply_exceptions` control table and the following warning:
Some changes from the source database had no impact when applied to the target database. See `attrep_apply_exceptions` table for details.
- Columnstore tables without a Primary Key or Unique Index are not supported.
- UPDATES to a source column's data type or size will not be captured when the **Create columnstore tables** option is enabled [in the Advanced tab](#).



Such *UPDATEs* can be performed on the target using a global or table transformation.

- Replicating data with 4-byte emoji characters is not supported.

Permissions

You must provide SingleStore account access to the Qlik Replicate user. This user must have read/write privileges in the SingleStore database.

Additionally, for each target schema involved in the replication, the Qlik Replicate user must have the following permissions:

- ALTER
- CREATE
- DELETE
- INSERT
- SELECT
- UPDATE

Supported data types

The following table shows the SingleStore database target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.

For information on how to view the data type that is mapped from the source, see the section for the source endpoint you are using.

For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).

Supported data types

Qlik Replicate Data Types	SingleStore Data Types
BOOL	BOOL
BYTES	<p>If length is => 1 and =< 8095, then:</p> <p>VARBINARY (Length)</p> <p>If length is => 8096 and =< 65535, then:</p> <p>BLOB</p> <p>If length is => 65536 and =< 16777215, then:</p> <p>MEDIUMBLOB</p> <p>If length is => 16777216 and =< 2147483647, then:</p> <p>LOB</p>

10 Adding and managing target endpoints

Qlik Replicate Data Types	SingleStore Data Types
DATE	DATE
TIME	TIME
DATETIME	If scale is => 0 and =< 6, then: DECIMAL (p,s) If scale is => 7 and =< 12, then: VARCHAR (37)
INT1	TINYINT
INT2	SMALLINT
INT4	INTEGER
INT8	BIGINT
NUMERIC	If scale is => 0 and =< 30, then: DECIMAL (p,s) If scale is => 31 and =< 100, then: VARCHAR (45)
REAL4	FLOAT
REAL8	DOUBLE
STRING	If length is => 1 and =< 8095, then: VARCHAR (Length) If length is => 8096 and =< 65535, then: TEXT If length is => 65536 and =< 16777215, then: MEDIUMTEXT If length is => 16777216 and =< 2147483647, then: LONGTEXT
UINT1	UNSIGNED TINYINT
UINT2	UNSIGNED SMALLINT
UINT4	UNSIGNED INTEGER
UINT8	UNSIGNED BIGINT

Qlik Replicate Data Types	SingleStore Data Types
WSTRING	<p>If length is => 1 and =< 8095, then:</p> <p>VARCHAR (Length)</p> <p>If length is => 8096 and =< 65535, then:</p> <p>TEXT</p> <p>If length is => 65536 and =< 16777215, then:</p> <p>MEDIUMTEXT</p> <p>If length is => 16777216 and =< 2147483647, then:</p> <p>LONGTEXT</p>
BLOB	<p>When the Allow unlimited LOB size option is enabled:</p> <ul style="list-style-type: none"> • LONGBLOB <p>When the Limit LOB size to option is enabled:</p> <ul style="list-style-type: none"> • If the specified value i is 63 KB or smaller, then: BLOB • If the specified value is 64 KB or larger, then: LONGBLOB <p>For information on limiting LOB size, see <i>Target metadata (page 1166)</i>.</p>
NCLOB	<p>When the Allow unlimited LOB size option is enabled:</p> <ul style="list-style-type: none"> • TEXT <p>When the Limit LOB size to option is enabled:</p> <ul style="list-style-type: none"> • If the specified value is 63 KB or smaller, then: TEXT • If the specified value is 64 KB or larger, then: LONGTEXT <p>For information on limiting LOB size, see <i>Target metadata (page 1166)</i>.</p>

Qlik Replicate Data Types	SingleStore Data Types
CLOB	<p>When the Allow unlimited LOB size option is enabled:</p> <ul style="list-style-type: none"> • TEXT CHARACTER SET UTF-16 <p>When the Limit LOB size to option is enabled:</p> <ul style="list-style-type: none"> • If the specified value is 63 KB or smaller, then: TEXT CHARACTER SET UTF-16 • If the specified value is 64 KB or larger, then: LONGTEXT CHARACTER SET UTF-16 <p>For information on limiting LOB size, see <i>Target metadata (page 1166)</i>.</p>

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a SingleStore target endpoint to Qlik Replicate:

1. In the Qlik Replicate Console, click **Manage Endpoint Connections** to open the **Manage EndpointsConnections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the SingleStore database. This is optional.
4. Select **TARGET** as the database **role**.
5. From the **Type** drop-down list, select SingleStore.
6. In the **Server** field, enter the host name or IP address of the computer on which the SingleStore database is installed.



Notes

- *This information is case sensitive.*
- *To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**. If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box. To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.*

7. Optionally, change the default port (3306).

8. Enter the **Username** and **Password** of a user authorized to access the SingleStore database specified below. If you do not know this information, see your SingleStore database Administrator (DBA).
9. In the **Database** field, specify the target SingleStore database.

Setting advanced connection properties

In the **Advanced** tab, you can set advanced properties, define internal parameters, and view a summary of your settings.

- **Max file size (KB):** Select or type the maximum size (in KB) of a CSV file before it is loaded into the SingleStore target database. The default value is 32000 KB.
- Select **Create columnstore tables** to create the target tables in columnstore format (as opposed to the default rowstore format).
Rowstore format is typically used for highly concurrent OLTP and mixed OLTP/analytical workloads. Columnstore format is used primarily for analytical applications where the queries mainly involve aggregations over data sets that are too large to store in memory. For a more detailed explanation of the differences between rowstore and columnstore tables, refer to the SingleStore Help.
- **Additional ODBC connection properties:** Set any additional ODBC connection properties as necessary.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.45 Using Snowflake on AWS as a target

This section describes how to set up and use Snowflake on AWS as a target in a replication task. Snowflake on AWS is located in the cloud and is accessed through an Amazon Web Services (AWS) account.

Qlik Replicate first stages the source data either directly on Snowflake or in an Amazon S3 bucket, and then loads it into the relevant tables in the Snowflake data warehouse.



When configuring a task with a Snowflake on AWS target endpoint, note the following:

- If the [Apply Changes](#) task option is enabled and the change processing mode is set to [Batch optimized apply](#), it is recommended to enable the **Apply batched changes to multiple tables concurrently** option in the Change Processing Tuning (page 1181) tab.
- When the **Apply batched changes to multiple tables concurrently** option is selected, the option to set a Global Error Handling policy will not be available. Also, some of the task-specific error handling defaults will be different.
For more information, see the **Apply batched changes to multiple tables concurrently** option in the Change Processing Tuning (page 1181) tab.

In this section:

- [Prerequisites](#) (page 1021)
- [Limitations and considerations](#) (page 1022)
- [Required permissions](#) (page 1032)
- [Snowflake on AWS data types](#) (page 1023)
- [Setting general connection parameters](#) (page 1025)
- [Setting advanced connection properties](#) (page 1027)

Prerequisites

The following describes the general prerequisites for using Snowflake on AWS as a Qlik Replicate target endpoint.

Amazon Web Services account prerequisites

Sign up for an Amazon Web Services account. Then use the AWS Management Console to purchase [Snowflake on AWS On Demand - Standard](#) or [Snowflake on AWS On Demand - Premier](#) and launch a Snowflake on AWS cluster. After registering for an Amazon Web Services (AWS) account, you can launch a Snowflake on AWS cluster and download the required client tools.

Make a note of the basic information about your AWS account and your Snowflake on AWS cluster, such as your password and user name. You will need this information to configure Qlik Replicate to work with the Snowflake on AWS data warehouse. For more information, see *Setting general connection parameters (page 1025)*.

See also:

[Snowflake Documentation](#)

Amazon S3 staging prerequisites

You can configure the Snowflake on AWS endpoint to stage the data files on Snowflake (internally) or on Amazon S3. If you want to use Amazon S3 staging, you need to have an Amazon S3 bucket, preferably (for best performance) located in your Snowflake on AWS cluster region.

You must be able to access your Amazon S3 bucket directly from the Replicate machine.

For information on signing up for Amazon S3, visit:

<https://aws.amazon.com/console/>

- **Bucket access credentials:** Make a note of the bucket name, region, access key and secret access key - you will need to provide them in the Qlik Replicate Snowflake on AWS target settings.
- **Bucket access permissions:** Qlik Replicate requires read/write/delete permissions to the Amazon S3 bucket.

Client prerequisites

Qlik Replicate for Windows:

Download and install Snowflake ODBC driver 2.24.0 (64-bit) or later for Windows on the Qlik Replicate Server machine.

Qlik Replicate for Linux:

Download and install Snowflake ODBC driver 2.24.0 (64-bit) or later for Linux on the Qlik Replicate Server machine.

Performance and cloud services usage optimization

In order to optimize both Snowflake Cloud Services usage and overall performance, it is strongly recommended to enable the [Apply batched changes to multiple tables concurrently](#) option in the task settings' **Change Processing Tuning** tab.

For information about charges incurred when using Snowflake Cloud Services, see [Snowflake: Additional Billing \(Cloud Services Layer\) Impact](#)

General prerequisites

Make sure the time and time zone settings on the Replicate machine are accurate. This is required in order to ensure:

- Proper synchronization of Full Load and CDC tasks.
- Correlation of the transaction log time with the actual time.

Firewall prerequisites

Firewall port 443 needs to be opened for outbound communication.

Limitations and considerations

The following limitations apply when using the Snowflake on AWS target endpoint in a Replicate task:

- The **Transactional apply** [Change Processing mode](#) is not supported.
- Replication of tables with backslashes in their names (e.g. SPECIAL\\TABLE\N) is not supported.

Required permissions

This topic describes the permissions required for working with this endpoint.

Setting a custom role

When Replicate establishes an ODBC connection to Snowflake, it uses the default user role on Snowflake, which is SYSADMIN.

However, if your corporate security policy dictates that third-part applications should not have SYSADMIN privileges, it is possible to create a custom role with fewer privileges and use that instead.

This can be done by changing the default user role on Snowflake from SYSADMIN to the desired custom role.

This can be done using either of the following methods:

- In the **Advanced** tab's **Additional ODBC connection properties** field, specify `role=MYROLE` where `MYROLE` is a custom role defined on Snowflake.
- Change the default user role on Snowflake from SYSADMIN to the desired custom role.

Required permissions

The required permissions differ according to whether or not the schema and/or the target tables already existed before the Replicate task started.

Tables created by Replicate

Permissions required if you want schemas that do not exist to be created automatically

- USAGE ON DATABASE
- CREATE SCHEMA ON DATABASE

Permissions required if the schema already exists

- USAGE ON DATABASE
- USAGE ON SCHEMA
- CREATE FILE FORMAT ON SCHEMA
- CREATE STAGE ON SCHEMA
- CREATE TABLE ON SCHEMA

Tables that existed before the Replicate task started

Although not recommended, if you wish data to be replicated to existing tables *that were created by a user with a different role*, the Snowflake user specified in the endpoint settings must have the following table privileges:

- SELECT
- INSERT
- UPDATE
- TRUNCATE
- DELETE
- REFERENCES

Snowflake on AWS data types

The Snowflake on AWS endpoint for Qlik Replicate supports most Snowflake data types. The following table shows the Snowflake on AWS target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.



*Snowflake on AWS does not support applying changes to binary data types in **Batch optimized apply** mode. For more information on **Batch optimized apply** mode, see *Change Processing Tuning* (page 1181).*

For information on how to view the data type that is mapped from the source, see the section for the source database you are using. For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).

Supported data types

Qlik Replicate Data Types	Snowflake on AWS Data Types
BOOL	BOOLEAN

10 Adding and managing target endpoints

Qlik Replicate Data Types	Snowflake on AWS Data Types
BYTES	<p>If length is => 1 and =< 8388608, then:</p> <p>BINARY (Length in Bytes)</p> <p>If length is => 8388609 and =< 2147483647, then:</p> <p>BINARY (8388608)</p>
DATE	DATE
TIME	TIME
DATETIME	TIMESTAMP (precision)
INT1	NUMBER
INT2	NUMBER
INT4	NUMBER
INT8	NUMBER
NUMERIC	<p>If scale is => 0 and =< 37, then:</p> <p>NUMBER (p,s)</p> <p>If scale is => 38 and =< 127, then:</p> <p>NUMBER (Length)</p>
REAL4	FLOAT4
REAL8	FLOAT8
UINT1	BYTEINT
UINT2	INTEGER
UINT4	INTEGER
UINT8	BIGINT
STRING	<p>If length is => 1 and =< 16777216, then:</p> <p>VARCHAR (Length in Bytes)</p> <p>If length is => 16777217 and =< 2147483647, then:</p> <p>VARCHAR (16777216)</p> <p>If subtype is JSON or XML, then:</p> <p>VARIANT</p>

Qlik Replicate Data Types	Snowflake on AWS Data Types
WSTRING	<p>If length is => 1 and =< 21845, then:</p> <p>VARCHAR (Length in Bytes)</p> <p>If length is => 21846 and =< 2147483647, then:</p> <p>VARCHAR (65535)</p> <p>If subtype is JSON or XML, then:</p> <p>VARIANT</p>
BLOB	<p>If length is => 1 and =< 8388608, then:</p> <p>BINARY (Length in Bytes)</p> <p>If length is => 8388609 and =< 2147483647, then:</p> <p>BINARY (8388608)</p>
NCLOB	<p>NVARCHAR (Max LOB Size)</p> <p>If subtype is JSON or XML, then:</p> <p>VARIANT</p>
CLOB	<p>VARCHAR (Max LOB Size)</p> <p>If subtype is JSON or XML, then:</p> <p>VARIANT</p>



Full LOB data types are not supported. For information on including Limited-size LOB data types in the replication, see the Metadata (page 1166) tab description in Customizing tasks (page 1082).

Setting general connection parameters

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a Snowflake on AWS target endpoint to Qlik Replicate:

1. In the Qlik Replicate Console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box.
2. In the **Manage Endpoint Connections** dialog box, click **New Endpoint Connection**.
3. In the **Name** field, specify a name for your Snowflake on AWS endpoint.
4. Optionally, in the **Description** field, enter a description for the Snowflake on AWS target endpoint.
5. Select **Target** as the **role**.
6. Select **Snowflake on AWS** as the **Type**.

7. Configure the remaining settings in the **General** tab as described in the table below.

- **Snowflake on AWS Target:**

- **Snowflake Account/Host:** Your URL for accessing Snowflake on AWS.
- **User name:** Your Snowflake user name.
- **Password:** The password for the user entered in the **User name** field.
- **Warehouse:** The name of your Snowflake warehouse.
- **Database name:** The name of your Snowflake database.

- **Staging** - Choose either **Snowflake** or **AWS S3**.

If you selected **AWS S3**, provide the following information:

- **Bucket name:** The name of the Amazon S3 bucket to where the files will be copied.

- **Bucket region:**

The region where your bucket is located. It is recommended to leave the default (**Auto-Detect**) as it usually eliminates the need to select a specific region. However, due to security considerations, for some regions (for example, AWS GovCloud) you might need to explicitly select the region. If the region you require does not appear in the regions list, select **Other** and specify the code in the **Region code** field.

For a list of region codes, see the **Region availability** section in:

<https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Concepts.RegionsAndAvailabilityZones.html>

- **Access type:** Choose one of the following:

- **Key pair** - Choose this method to authenticate with your Access Key and Secret Key. Then provide the following additional information:

- **Access key:** Type the access key information for Amazon S3.
- **Secret key:** Type the secret key information for Amazon S3.

- **IAM Roles for EC2** - Choose this method if the machine on which Qlik Replicate is installed is configured to authenticate itself using an IAM role. Then provide the following additional information:

- **External stage name:** The name of your external stage. To use the IAM Roles for EC2 access type, you must create an external stage that references the S3 bucket.

To use the IAM Roles for EC2 access method, you also need to fulfill the prerequisites described in *Prerequisite for using the IAM Roles for EC2 Access Type* (page 1026).

- **Folder:** The bucket folder to where the files will be copied.



The information for these properties is available from the account page for Amazon Web Services (AWS) with the Snowflake on AWS cluster. If you do not have these values, refer to your AWS account or the Snowflake on AWS System Administrator for your enterprise.

Prerequisite for using the IAM Roles for EC2 Access Type

To use the IAM Roles for EC2 access type, you must run the following commands on the Snowflake on AWS database before running the task:

10 Adding and managing target endpoints

Command 1:

```
create or replace file format MY_FILE_FORMAT TYPE='CSV' field_delimiter=', 'compression='GZIP'  
record_delimiter='\n' null_if=('attrep_null') skip_header=0 FIELD_OPTIONALLY_ENCLOSED_BY='\';
```

Command 2:

```
create or replace stage "PUBLIC".MY_S3_STAGE file_format=MY_FILE_FORMAT url='s3://MY_STORAGE_  
URL' credentials=(aws_role='MY_IAM_ROLE');
```

Where:

MY_FILE_FORMAT - Can be any value.

MY_S3_STAGE - The name specified in the **External stage name** field above.

MY_STORAGE_URL - The URL of your Amazon S3 bucket

MY_IAM_ROLE - Your IAM role name.



To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**.

If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

Setting advanced connection properties

In the **Advanced** tab, you can set the parameters described in the table below.

General properties

Option	Description
Max file size (MB)	Select or type the maximum size of the CSV file used to transfer data to Snowflake. The default value is 100 MB.
ODBC driver	The name of the default ODBC driver you are using to connect to Snowflake. The default value is SnowflakeDSIIDriver .
Additional ODBC connection properties	Type any additional ODBC connection properties if required.

10 Adding and managing target endpoints

Proxy server properties

Option	Description
Use proxy server	Select this option to access Amazon S3 via a proxy server.
Host name	The host name of the proxy server.
Port	The port via which to access the proxy server.
User name	The user name for accessing the proxy server.
Password	The password for accessing the proxy server.
Scheme	<p>Select which protocol to use to access the server (HTTP or HTTPS). In order to use HTTPS, you must first install the CA certificate that signed the proxy's certificate on the Replicate Server machine. Do this as follows:</p> <ul style="list-style-type: none">• On Windows: Add the CA certificate to the Trusted Root Certification Authorities store of Local Computer• On Linux: Add the CA certificate to <code>/etc/pki/tls/certs/ca-bundle.crt</code> <p>The HTTPS proxy scheme is supported for AWS S3 staging only. It is not supported for Snowflake staging or Snowflake Access (ODBC).</p> <p>See also: <i>Setting general connection parameters (page 1025)</i>.</p>

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.46 Using Snowflake on Azure as a target

This section describes how to set up and use Snowflake on Azure as a target in a replication task.



When configuring a task with a Snowflake on Azure target endpoint, note the following:

- If the [Apply Changes](#) task option is enabled and the change processing mode is set to [Batch optimized apply](#), it is recommended to enable the **Apply batched changes to multiple tables concurrently** option in the Change Processing Tuning (page 1181) tab.
- When the **Apply batched changes to multiple tables concurrently** option is selected, the option to set a Global Error Handling policy will not be available. Also, some of the task-specific error handling defaults will be different.
For more information, see the **Apply batched changes to multiple tables concurrently** option in the Change Processing Tuning (page 1181) tab.

In this section:

- [Prerequisites](#) (page 1031)
- [Limitations and considerations](#) (page 1032)
- [Required permissions](#) (page 1032)
- [Snowflake on Azure data types](#) (page 1033)
- [Setting general connection parameters](#) (page 1035)
- [Setting advanced connection properties](#) (page 1036)

Prerequisites

The following section describes the prerequisites for using Snowflake on Azure as a Qlik Replicate target endpoint.

Client prerequisites

Qlik Replicate for Windows:

Download and install Snowflake ODBC driver 2.24.0 (64-bit) or later for Windows on the Qlik Replicate Server machine.

Qlik Replicate for Linux:

Download and install Snowflake ODBC driver 2.24.0 (64-bit) or later for Linux on the Qlik Replicate Server machine.

Performance and Cloud Services usage optimization

In order to optimize both Snowflake Cloud Services usage and overall performance, it is strongly recommended to enable the [Apply batched changes to multiple tables concurrently](#) option in the task settings' **Change Processing Tuning** tab.

For information about charges incurred when using Snowflake Cloud Services, see [Snowflake: Additional Billing \(Cloud Services Layer\) Impact](#)

Microsoft Azure Blob Storage prerequisites

You can configure the Snowflake on Azure endpoint to stage the data files on Snowflake (Windows and Linux) or on Azure Blob storage (Windows only). If you want to use Azure Blob storage, you need to sign up for a Microsoft Azure Blob Storage account and make a note of the account name, access key, container name, SAS token (Shared Access Signature), and target folder.

Note that the duration of the SAS token must be the same as the Replicate task duration. For an explanation of how to configure the SAS token, visit:

<https://docs.snowflake.com/en/user-guide/data-load-azure-config.html>

For information on signing up for a Microsoft Azure Blob Storage account, see the vendor's online help.

Microsoft Azure Blob Storage permissions

Qlik Replicate performs the following operations on the Azure Blob Storage container/folder:

- On the Azure Blob Storage container: LIST and CREATE on SCHEMA public
- On the Azure Blob Storage folder: READ, WRITE, and DELETE

Supported blob storage types

The following blob storage types are supported:

- Standard storage with block blobs
- Premium storage with block blobs only

Firewall prerequisites

Firewall port 443 needs to be opened for outbound communication.

Limitations and considerations

The following limitations apply when using the Snowflake on Azure target endpoint in a Replicate task:

- The **Transactional apply** [Change Processing mode](#) is not supported.
- Replication of tables with backslashes in their names (e.g. SPECIAL\\TABLE\N) is not supported.

Required permissions

This topic describes the permissions required for working with this endpoint.

Setting a custom role

When Replicate establishes an ODBC connection to Snowflake, it uses the default user role on Snowflake, which is SYSADMIN.

However, if your corporate security policy dictates that third-part applications should not have SYSADMIN privileges, it is possible to create a custom role with fewer privileges and use that instead.

This can be done by changing the default user role on Snowflake from SYSADMIN to the desired custom role.

This can be done using either of the following methods:

- In the **Advanced** tab's **Additional ODBC connection properties** field, specify `role=MYROLE` where `MYROLE` is a custom role defined on Snowflake.
- Change the default user role on Snowflake from SYSADMIN to the desired custom role.

Required permissions

The required permissions differ according to whether or not the schema and/or the target tables already existed before the Replicate task started.

Tables created by Replicate

Permissions required if you want schemas that do not exist to be created automatically

- USAGE ON DATABASE
- CREATE SCHEMA ON DATABASE

Permissions required if the schema already exists

- USAGE ON DATABASE
- USAGE ON SCHEMA
- CREATE FILE FORMAT ON SCHEMA
- CREATE STAGE ON SCHEMA
- CREATE TABLE ON SCHEMA

Tables that existed before the Replicate task started

Although not recommended, if you wish data to be replicated to existing tables *that were created by a user with a different role*, the Snowflake user specified in the endpoint settings must have the following table privileges:

- SELECT
- INSERT
- UPDATE
- TRUNCATE
- DELETE
- REFERENCES

Snowflake on Azure data types

Qlik Replicate Snowflake on Azure endpoint supports most Snowflake data types. The following table shows the Snowflake target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.



The Snowflake on Azure endpoint does not support applying changes to binary data types in **Batch optimized apply** mode. For more information on **Batch optimized apply** mode, see *Change Processing Tuning* (page 1181).

For information on how to view the data type that is mapped from the source, see the section for the source database you are using. For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).

Replicate and Snowflake on Azure data types

Qlik Replicate Data Types	Snowflake on Azure Data Types
BOOL	BOOLEAN
BYTES	If length is => 1 and =< 8388608, then: BINARY (Length in Bytes) If length is => 8388609 and =< 2147483647, then: BINARY (8388608)
DATE	DATE
TIME	TIME
DATETIME	TIMESTAMP (precision)
INT1	NUMBER
INT2	NUMBER
INT4	NUMBER

10 Adding and managing target endpoints

Qlik Replicate Data Types	Snowflake on Azure Data Types
INT8	NUMBER
NUMERIC	<p>If scale is \Rightarrow 0 and \Leftarrow 37, then:</p> <p>NUMBER (p,s)</p> <p>If scale is \Rightarrow 38 and \Leftarrow 127, then:</p> <p>NUMBER (Length)</p>
REAL4	FLOAT4
REAL8	FLOAT8
UINT1	BYTEINT
UINT2	INTEGER
UINT4	INTEGER
UINT8	BIGINT
STRING	<p>If length is \Rightarrow 1 and \Leftarrow 16777216, then:</p> <p>VARCHAR (Length in Bytes)</p> <p>If length is \Rightarrow 16777217 and \Leftarrow 2147483647, then:</p> <p>VARCHAR (16777216)</p> <p>If subtype is JSON or XML, then:</p> <p>VARIANT</p>
WSTRING	<p>If length is \Rightarrow 1 and \Leftarrow 21845, then:</p> <p>VARCHAR (Length in Bytes)</p> <p>If length is \Rightarrow 21846 and \Leftarrow 2147483647, then:</p> <p>VARCHAR (65535)</p> <p>If subtype is JSON or XML, then:</p> <p>VARIANT</p>
BLOB	<p>If length is \Rightarrow 1 and \Leftarrow 8388608, then:</p> <p>BINARY (Length in Bytes)</p> <p>If length is \Rightarrow 8388609 and \Leftarrow 2147483647, then:</p> <p>BINARY (8388608)</p>

Qlik Replicate Data Types	Snowflake on Azure Data Types
NCLOB	VARCHAR (Max LOB Size) If subtype is JSON or XML, then: VARIANT
CLOB	VARCHAR (Max LOB Size) If subtype is JSON or XML, then: VARIANT



Full LOB data types are not supported. For information on including Limited-size LOB data types in the replication, see the Metadata (page 1166) tab description in Customizing tasks (page 1082).

Setting general connection parameters

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a Snowflake on Azure target endpoint to Qlik Replicate:

1. In the Qlik Replicate Console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box.
2. In the **Manage Endpoint Connections** dialog box, click **New Endpoint Connection**.
3. In the **Name** field, type a name for your Snowflake on Azure endpoint.
4. Optionally, in the **Description** field, enter a description for your Snowflake on Azure target endpoint.
5. Select **Target** as the **role**.
6. Select **Snowflake on Azure** as the **Type**.
7. Configure the remaining settings in the **General** tab as described below.
 - **Snowflake on Azure Target:**
 - **Snowflake Account/Host:** Your URL for accessing Snowflake on Microsoft Azure.
 - **User name:** Your Snowflake user name.
 - **Password:** The password for the user entered in the **User name** field.
 - **Warehouse:** The name of your Snowflake warehouse.
 - **Database name:** The name of your Snowflake database.
 - **Staging:**
 - **Storage Type:** Choose one of the following:
 - **Azure Blob storage:** If you select this option, you also need to provide the access and storage information specified below.
 - **Snowflake:** When this option is selected, Snowflake's internal storage will be used.
 - **Storage account:** The name of an account with write permissions to the container.



To connect to an Azure resource on Government Cloud or China Cloud, you need to specify the full resource name of the storage account. For example, assuming the storage account is `MyBlobStorage`, then the resource name for China cloud would be `MyBlobStorage.dfs.core.chinacloudapi.cn`

For information on setting internal parameters, see [Setting advanced connection properties](#) (page 1036)

- **Access key:** The account access key.
 - **SAS token:** Your SAS (Shared Access Signature) for accessing the container.
 - **Container name:** The container name.
 - **Folder:** The container folder to where the files will be copied.



To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**.

If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

Setting advanced connection properties

In the **Advanced** tab, you can set the parameters described in the table below.

Advanced connection properties

Option	Description
Max file size (MB)	Select or type the maximum size of the CSV file used to transfer data to Snowflake on Azure. The default value is 100 MB.
ODBC driver	The name of the default ODBC driver you are using to connect to Snowflake on Azure. The default value is SnowflakeDSIIDriver .
Additional ODBC connection properties	Type any additional ODBC connection properties if required.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.47 Using Snowflake on Google as a target

This section describes how to set up and use Snowflake on Google as a target in a replication task.



When configuring a task with a Snowflake on Google target endpoint, note the following:

- If the [Apply Changes](#) task option is enabled and the change processing mode is set to [Batch optimized apply](#), it is recommended to enable the **Apply batched changes to multiple tables concurrently** option in the Change Processing Tuning (page 1181) tab.
- When the **Apply batched changes to multiple tables concurrently** option is selected, the option to set a Global Error Handling policy will not be available. Also, some of the task-specific error handling defaults will be different.
For more information, see the **Apply batched changes to multiple tables concurrently** option in the Change Processing Tuning (page 1181) tab.

In this section:

- [Prerequisites](#) (page 1039)
- [Required permissions](#) (page 1040)
- [Limitations and considerations](#) (page 1039)
- [Snowflake on Google data types](#) (page 1041)
- [Setting general connection parameters](#) (page 1043)
- [Setting advanced connection properties](#) (page 1044)

Prerequisites

The following section describes the prerequisites for using Snowflake on Google as a Qlik Replicate target endpoint.

Client prerequisites

Qlik Replicate for Windows:

Download and install Snowflake ODBC driver 2.24.0 (64-bit) or later for Windows on the Qlik Replicate Server machine.

Qlik Replicate for Linux:

Download and install Snowflake ODBC driver 2.24.0 (64-bit) or later for Linux on the Qlik Replicate Server machine.



The expected name for the ODBC driver is **SnowflakeDSIIDriver** (the default).

Performance and Cloud Services usage optimization

In order to optimize both Snowflake Cloud Services usage and overall performance, it is strongly recommended to enable the [Apply batched changes to multiple tables concurrently](#) option in the task settings' **Change Processing Tuning** tab.

For information about charges incurred when using Snowflake Cloud Services, see [Snowflake: Additional Billing \(Cloud Services Layer\) Impact](#)

Storage integration name

You need to specify a storage integration name in the Snowflake on Google target endpoint settings. Integrations avoid the need for passing explicit cloud provider credentials such as secret keys or access tokens; instead, integration objects reference a Cloud Storage service account.

For more information on creating a storage integration name, see [Configuring an Integration for Google Cloud Storage](#)

Firewall prerequisites

Firewall port 443 needs to be opened for outbound communication.

Limitations and considerations

The following limitations apply when using the Snowflake target endpoint in a Replicate task:

- The [Transactional apply Change Processing mode](#) is not supported.
- Applying changes to binary data types is not supported.
- Supports working in [Limited LOB mode](#) only.
- Connecting to a proxy server via HTTPS is not supported.
- Replication of data with 4-byte emoji characters is not supported.



This limitation does not apply to Replicate November 2022 Service Release 1 or later.

- Replication of tables with backslashes in their names (e.g. SPECIAL\\TABLE\N) is not supported.

Required permissions

This topic describes the permissions required for working with this endpoint.

Setting a custom role

When Replicate establishes an ODBC connection to Snowflake, it uses the default user role on Snowflake, which is SYSADMIN.

However, if your corporate security policy dictates that third-part applications should not have SYSADMIN privileges, it is possible to create a custom role with fewer privileges and use that instead.

This can be done by changing the default user role on Snowflake from SYSADMIN to the desired custom role.

This can be done using either of the following methods:

Required permissions

The required permissions differ according to whether or not the schema and/or the target tables already existed before the Replicate task started.

Tables created by Replicate

Permissions required if you want schemas that do not exist to be created automatically

- USAGE ON DATABASE
- CREATE SCHEMA ON DATABASE

Permissions required if the schema already exists

- USAGE ON DATABASE
- USAGE ON SCHEMA
- CREATE FILE FORMAT ON SCHEMA
- CREATE STAGE ON SCHEMA
- CREATE TABLE ON SCHEMA

Tables that existed before the Replicate task started

Although not recommended, if you wish data to be replicated to existing tables *that were created by a user with a different role*, the Snowflake user specified in the endpoint settings must have the following table privileges:

- SELECT
- INSERT
- UPDATE

- TRUNCATE
- DELETE
- REFERENCES

Google cloud storage

The JSON credentials that you need to specify in the Snowflake on Google target endpoint settings must be for an account that has read and write access to the specified bucket and folder.

Snowflake on Google data types

Qlik Replicate Snowflake on Google endpoint supports most Snowflake data types. The following table shows the Snowflake target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.



- *The Snowflake on Google endpoint does not support applying changes to binary data types in **Batch optimized apply** mode. For more information on **Batch optimized apply** mode, see [Change Processing Tuning \(page 1181\)](#).*
- *The `INT`, `INTEGER`, `BIGINT`, `SMALLINT`, `TINYINT`, `BYTEINT` data types will always be represented as `NUMBER(38, 0)` on Snowflake. For more information, see the [Snowflake Help](#).*

For information on how to view the data type that is mapped from the source, see the section for the source database you are using. For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

Replicate and Snowflake data types

Qlik Replicate Data Types	Snowflake on Google Data Types
BOOL	BOOLEAN
BYTES	If length is => 1 and =< 8388608, then: BINARY (Length in Bytes) If length is => 8388609 and =< 2147483647, then: BINARY (8388608)
DATE	DATE
TIME	TIME
DATETIME	TIMESTAMP (precision)
INT1	NUMBER
INT2	NUMBER
INT4	NUMBER
INT8	NUMBER

10 Adding and managing target endpoints

Qlik Replicate Data Types	Snowflake on Google Data Types
NUMERIC	<p>If scale is \Rightarrow 0 and \leq 37, then:</p> <p>NUMBER (p,s)</p> <p>If scale is \Rightarrow 38 and \leq 127, then:</p> <p>NUMBER (Length)</p>
REAL4	FLOAT4
REAL8	FLOAT8
UINT1	BYTEINT
UINT2	INTEGER
UINT4	INTEGER
UINT8	BIGINT
STRING	<p>If length is \Rightarrow 1 and \leq 16777216, then:</p> <p>VARCHAR (Length in Bytes)</p> <p>If length is \Rightarrow 16777217 and \leq 2147483647, then:</p> <p>VARCHAR (16777216)</p> <p>If subtype is JSON or XML, then:</p> <p>VARIANT</p>
WSTRING	<p>If length is \Rightarrow 1 and \leq 21845, then:</p> <p>VARCHAR (Length in Bytes)</p> <p>If length is \Rightarrow 21846 and \leq 2147483647, then:</p> <p>VARCHAR (65535)</p> <p>If subtype is JSON or XML, then:</p> <p>VARIANT</p>
BLOB	<p>If length is \Rightarrow 1 and \leq 8388608, then:</p> <p>BINARY (Length in Bytes)</p> <p>If length is \Rightarrow 8388609 and \leq 2147483647, then:</p> <p>BINARY (8388608)</p>

Qlik Replicate Data Types	Snowflake on Google Data Types
NCLOB	VARCHAR (Max LOB Size) If subtype is JSON or XML, then: VARIANT
CLOB	VARCHAR (Max LOB Size) If subtype is JSON or XML, then: VARIANT



Full LOB data types are not supported. For information on including Limited-size LOB data types in the replication, see the Metadata (page 1166) tab description in Customizing tasks (page 1082).

Setting general connection parameters

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a Snowflake on Google target endpoint to Qlik Replicate:

1. In the Qlik Replicate Console, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box.
2. In the **Manage Endpoint Connections** dialog box, click **New Endpoint Connection**.
3. In the **Name** field, enter a display name for your Snowflake on Google endpoint.
4. Optionally, in the **Description** field, enter a description for your Snowflake on Google target endpoint.
5. Select **Target** as the **role**.
6. Select **Snowflake on Google** as the **Type**.
7. Configure the remaining settings in the **General** tab as described in the tables below.

Snowflake Access configuration

Option	Description
Snowflake Account/Host	Your URL for accessing Snowflake on Google.
User name	Your Snowflake user name.
Password	The password for the user entered in the User name field.
Warehouse	The name of your Snowflake warehouse.
Database name	The name of your Snowflake database.

10 Adding and managing target endpoints

Staging on Google Cloud Storage configuration

Option	Description
JSON credentials	The JSON credentials for the service account key with read and write access to the Google Cloud Storage bucket.
Bucket name	The Google Cloud Storage bucket.
Staging folder	Where to create the data files in the specified bucket.
Storage integration name	Your storage integration name. Integrations avoid the need for passing explicit cloud provider credentials such as secret keys or access tokens; instead, integration objects reference a Cloud Storage service account. For more information on creating a storage integration name, see https://docs.snowflake.com/en/user-guide/data-load-gcs-config.html



To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**.

If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.

To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

Setting advanced connection properties

In the **Advanced** tab, you can set the parameters described in the tables below.

File Attributes parameters

Option	Description
Max file size (MB)	Select or type the maximum size of the CSV file used to transfer data to Snowflake on Google. The default value is 100 MB.

Proxy Server parameters

Option	Description
Use proxy server	Select this option to access Snowflake on Google via a proxy server.
Host name	The host name of the proxy server.
Port	The port via which to access the proxy server.
User name	The user name for accessing the proxy server, if required.
Password	The password for accessing the proxy server, if required.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.48 Using Teradata Database as a target

This section explains how to set up and use Teradata Database as a target endpoint in a replication task.



Although the information in this section refers to Teradata Database, it is relevant for Teradata Vantage as well.

In this section:

- *An overview of the Teradata Database target (page 1047)*
- *Limitations and considerations (page 1047)*
- *Teradata Database target load options (page 1047)*
- *Database availability (page 1047)*
- *Prerequisites (page 1047)*
- *Providing access to the Teradata Database (page 1049)*
- *Security requirements (page 1050)*
- *Teradata Database data types (page 1050)*
- *Setting general connection properties (page 1053)*
- *Setting advanced connection properties (page 1054)*

An overview of the Teradata Database target

The Qlik Replicate database for Teradata Database is a powerful operational data warehousing solution that manages Big Data analytics and challenges. Qlik Replicate uses the Teradata Database Parallel Transporter (TPT) API to facilitate data loading. The ODBC API is used for other purposes such as metadata queries (DDL requests) and retrieving information from Teradata Database error tables.

Qlik Replicate for Teradata Database uses the TPT load to bulk load data into a Teradata Database target database. You can replicate data to the Teradata Database from any source database supported by Qlik Replicate. In addition, Qlik Replicate can replicate data from any source database that supports ODBC.

Limitations and considerations

Replicating data with 4-byte emoji characters is not supported.

Teradata Database target load options

You can apply changes in one of two modes:

- *TPT stream mode (page 1047)*
- *TPT load mode (page 1047)*

TPT stream mode

When using the TPT stream mode, the TPT Stream operator uses the Teradata Database TPump protocol to perform high-speed DML transactions in a near-real-time mode on tables. The TPT STREAM operator is less restrictive than the LOAD operator.

This mode lets tables be queried at the same time that a DML operation takes place.

TPT load mode

When using the TPT load mode, the TPT LOAD operator uses the Teradata Database FastLoad protocol to load a large volume of data at high speed into an empty table on the Teradata Database.

The TPT LOAD operator has some restrictions that include the following:

- The target table must be empty.
- The target table cannot have secondary indexes defined.

Database availability

Teradata Database with the tables that are being used for replication must be available to the system. This can be installed on any computer in your network.

For more information about the requirements for working with Qlik Replicate, see *Installing Qlik Replicate (page 39)*.

Prerequisites

The following describes the prerequisites necessary to prepare your environment to work with Qlik Replicate and Teradata Database.



Teradata Database must be installed in your network and be reachable from the computer where Qlik Replicate is installed.

Replicate server for Windows

You must install the following components on the Qlik Replicate Server machine:

- Teradata Database ODBC Driver 17.20 for Windows.
Download the driver from <https://downloads.teradata.com/download/connectivity/odbc-driver/windows>
- Teradata Database Parallel Processor API (TPT API) with the load and Stream TPT operators version 17.20.

Replicate server for Linux

The following section describes the steps you need to perform to work with Qlik Replicate for Linux and Teradata Database as a target database in a Replicate task. Teradata Database Client requires the DataDirect ODBC driver manager (provided with Teradata Database Client).



A Replicate task cannot be defined with endpoints that use different ODBC Driver Managers. Teradata Database target is accessed using the DataDirect ODBC Driver Manager. With the exception of Oracle, Hadoop, File and Replicate Connect sources (which are not subject to the above limitation), all other source endpoints use the unixODBC Driver Manager.

To configure a task with a unixODBC source and a DataDirect target (e.g. Microsoft SQL Server to Teradata Database Target), you need to use the Replicate File Channel. For more information about setting up a task using the File Channel, see [Using the Qlik Replicate file channel \(page 1061\)](#).

1. Stop the Replicate service:

```
./areplicate stop
```

2. Optionally, confirm that the service has stopped:

```
./areplicate status
```

3. Install the following Teradata client components:

- Install Teradata Database ODBC Driver 17.20 for Linux.
Download the driver from <https://downloads.teradata.com/download/connectivity/odbc-driver/linux>
- Teradata Database Parallel Processor API (TPT API) with the Load and Stream TPT operators

4. Open the Teradata odbcinst.ini file:

```
cat $TD_CLIENT_DIR/odbc_64/odbcinst.ini
```

Then verify that it contains a definition for the Teradata ODBC client:

```
[ODBC DRIVERS]
```

```
Teradata=Installed
```

```
[Teradata]
```

```
Driver=/opt/teradata/client/17.20/odbc_64/lib/tdata.so
```

```
DriverODBCVer=3.51
```

5. Check that directory `/usr/lib64` contains symbolic links to the DataDirect driver manager shared libraries:

```
ll /usr/lib64/libodbc*.so
```

The output should look like this:

```
lrwxrwxrwx 1 root root 47 Oct 28 14:58 /usr/lib64/libodbcinst.so ->
/opt/teradata/client/ODBC_64/lib/libodbcinst.so
lrwxrwxrwx 1 root root 43 Oct 28 14:58 /usr/lib64/libodbc.so ->
/opt/teradata/client/ODBC_64/lib/libodbc.so
```

6. Add the Teradata Database name to the hosts file as described in *Editing the hosts file* (page 1049).
7. Run the following commands:

```
export LD_LIBRARY_PATH=$TD_CLIENT:/opt/attunity/replicate/lib:/lib64:/usr/lib64
export ODBCINI=/opt/teradata/client/17.20/odbc_64/odbc.ini
export AREP_ODBC_DRIVER_MANAGER=/opt/teradata/client/17.20/odbc_64/lib/libodbc.so
```
8. Start the Replicate service:

```
./areplicate start
```
9. Optionally confirm that the service has started:

```
./areplicate status
```

Providing access to the Teradata Database

The Qlik Replicate user who is working with the Teradata Database must be registered as a user in the Teradata Database. This is the user that is entered in the dialog box when *Setting general connection properties* (page 1053). You must grant Teradata Database access to this user before configuring the database in Qlik Replicate.

Editing the hosts file

To enable Qlik Replicate to access the Teradata Database, you need to add the Teradata Database machine IP/name and database mappings to the Windows/Linux hosts file.

To add the Teradata Database mappings to the hosts file:

1. Open the Windows/Linux hosts file on the Qlik Replicate machine.
On Windows, the default path for the hosts file is:

```
~:\windows\system32\drivers\etc\hosts
```


On Linux, the default path for the hosts file is:

```
/etc/hosts
```
2. Add the following line (note the “cop1” after the database name):

```
<Teradata Database IP address/hostname> <Teradata Database name>cop1
```

Example:

```
123.123.123.1 teradatadbonecop1
```



Make sure that the database name added to the hosts files is the same as the database specified in the **Default database** field in the Teradata Database target database settings.

3. Save your changes.

Security requirements

A user must have the following privileges granted in the Teradata Database to use a Teradata Database target in a Qlik Replicate task:

- GRANT SELECT ON <database>
 - GRANT INSERT ON <database>
 - GRANT DELETE ON <database>
 - GRANT UPDATE ON <database>
 - GRANT EXECUTE ON <database>
 - GRANT EXECUTE FUNCTION ON <database>
 - GRANT EXECUTE PROCEDURE ON <database>
 - GRANT CREATE TABLE ON <database>
 - GRANT DROP TABLE ON <database>
 - GRANT CREATE VIEW ON <database>
 - GRANT DROP VIEW ON <database>
 - GRANT NONTEMPORAL on <database>
 - GRANT CHECKPOINT ON <database>
- When the Stream TPT Operator is selected (in the **Advanced** tab), the following privilege is also required:
- GRANT CREATE MACRO ON <database>

Teradata Database data types

The Teradata Database target endpoint for Qlik Replicate supports most Teradata Database data types. The following table shows the Teradata Database target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types. Unsupported data types are listed below the table.




*Teradata Database does not support applying changes to binary data types in **Batch optimized apply** mode. For more information on **Batch optimized apply** mode, see *Change Processing Tuning* (page 1181).*



For information on how to view the data type that is mapped from the source, see the section for the source database you are using. For additional information about Qlik Replicate data types, see *Replicate data types* (page 81).

Supported data types mapping

Qlik Replicate Data Types	Teradata Database Data Types
BOOLEAN	BYTEINT

10 Adding and managing target endpoints

Qlik Replicate Data Types	Teradata Database Data Types
BYTES	VARBYTE (Size) <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;">  <p>Maximum size is 640000.</p> </div>
DATE	DATE
TIME	TIME (P)
DATETIME	TIMESTAMP (P)
INT1	BYTEINT
INT2	SMALLINT
INT4	INTEGER
INT8	BIGINT
NUMERIC	If scale is => 0 and <= 37, then: NUMERIC (p,s) If scale is => 38, then: VARCHAR (45)
REAL4	FLOAT FLOAT is equivalent to REAL and DOUBLE PRECISION.
REAL8	FLOAT
STRING See also the note in <i>Teradata Database data types (page 1050)</i> below.	VARCHAR (Size) Note: Maximum size is 64000.
UINT1	BYTEINT
UINT2	SMALLINT
UINT4	INTEGER
UINT8	BIGINT

Qlik Replicate Data Types	Teradata Database Data Types
<p>WSTRING</p> <p>See also the note in <i>Teradata Database data types (page 1050)</i> below.</p>	<p>VARCHAR (Size)</p> <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;">  <i>Maximum size is 640000.</i> </div>
<p>BLOB</p>	<p>VARBYTE({MAX_LOB_SIZE})</p> <p>MAX_LOB_SIZE is the maximum LOB size specified in Limited-Size LOB Mode.</p>
<p>CLOB</p> <div style="border: 1px solid gray; padding: 10px; margin-top: 10px;">  <i>By default, Replicate multiplies the value of each varchar column by three, in order to support NLS. For example, a varchar column with 36 characters in the source database will have 108 characters in Teradata Database. This may result in Teradata Database varchar columns being longer than you actually need them (and unnecessarily increasing the row size). In such cases, you can override the default multiplication factor by using the <code>n1sFactor</code> internal parameter. For instructions on using the <code>n1sFactor</code> parameter, contact Qlik Support.</i> </div>	<p>VARCHAR({MAX_LOB_SIZE})</p> <p>Unicode case-insensitive character set.</p> <p>MAX_LOB_SIZE is the maximum LOB size specified in Limited-Size LOB Mode.</p>
<p>NCLOB</p> <p>See the note in <i>CLOB (page 1052)</i> above.</p>	<p>VARCHAR({MAX_LOB_SIZE})</p> <p>Case-insensitive character set.</p> <p>MAX_LOB_SIZE is the maximum LOB size specified in Limited-Size LOB Mode.</p>



About Teradata Database LOB support:

*Full LOB data types are not supported in the Teradata Database. For information on including Limited-size LOB data types in the replication, see *Metadata (page 1166)*. Note also that the size of a row in the Teradata Database cannot exceed 64KB. This should be taken into consideration when specifying the maximum LOB size in the **Metadata** tab.*

*See also the note in *CLOB (page 1052)* above.*

The following Teradata Database data types are not supported:

- PERIOD

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add a Teradata Database Target to Qlik Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button. For more information on adding an endpoint to Qlik Replicate, see *Defining and managing endpoints (page 114)*.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Teradata Database. This is optional.
4. Select **Target** as the database **role**.
5. Select **Teradata Database** as the database **Type**.
6. Type the **Server** name. This is the name of the computer with the Teradata Database instance you want to work with.
7. Type the Teradata Database authentication information (**Username, Password**) for the authorized user for this Teradata Database. If you do not know this information, see your Teradata Database system manager.



Consider the following:

- If you are using the **Advanced** tab to create a custom string, make sure to include the **USERNAME** property. A **Password** can also be included but is not required. See *Setting advanced connection properties (page 1054)* for more information.
- This information is case sensitive.
- If you want to set custom properties for this database, see *Setting advanced connection properties (page 1054)*.
- To determine if you are connected to the database you want to use or if the connection information you entered is correct, click **Test Connection**. If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box. To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.



Make sure that the Teradata Database user entered in the Teradata Database Authentication section has the correct access privileges. For information on how to provide the required privileges, see *Security requirements (page 1050)*.

8. Type the **Default database** name or select one from the list of available endpoints. This is the name of the Teradata Database where you are replicating the data to.
For more information, see *Teradata Database target load options (page 1047)*.

Setting advanced connection properties

You can set custom properties or change the default settings for various parameters by adding them to a custom connect string in the **Advanced** tab of the Add Database dialog box.

You can set the following parameters:

- **TPT Operator:** Select the TPT Operator used to access the Teradata Database. The possible options are:
 - **Load:** Select this to use the *TPT load mode (page 1047)*.
 - **Stream:** Select this to use the *TPT stream mode (page 1047)*.See *Teradata Database target load options (page 1047)*.
- **TPT Attributes:** You can define one or more of the following attributes:
 - **Account String:** The account (database server login) that the DBA assigned to the username for the Qlik Replicate user.
 - **Buffer Size:** The output buffer size (in KB) for sending Load parcels to the Teradata Database. You can enter a value from 1 to 64.
 - **Buffers:** The number of request buffers used. You can enter any value from 2 or higher.
 - **Explicit sessions range:** Select this if you want set a minimum and/or maximum number of sessions that can log on to the Teradata Database.
 - **Maximum:** The maximum number of sessions that can log on to the Teradata Database. The default value is 1. The value cannot be higher than the number of Access Module Processors (AMPs) available.
 - **Minimum:** The minimum number of sessions that can log on to the Teradata Database.
 - **Dynamic statement packing:** Select this check box if you want the stream driver to dynamically determine the maximum possible pack for the current STREAM job.
 - **Statement packing:** Use the counter or type the number of statements that can be packed into a multiple-statement request (STREAM). You can enter a value from 2 to 600. This is available only if **Statement packing** is not selected.
 - **Additional ODBC connection properties:** Type any additional ODBC connection properties, if required.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

10.49 Using Vertica as a target

This section describes how to set up and use an Vertica database as a target database in a replication task.

In this section:

- *Prerequisites (page 1057)*
- *Limitations (page 1057)*
- *Security requirements (page 1057)*
- *Vertica target data types (page 1058)*
- *Setting general connection properties (page 1059)*
- *Setting advanced connection properties (page 1060)*

Prerequisites

The following section describes the prerequisites for working with Qlik Replicate on Windows or Linux and the Vertica target database.

Replicate server for Windows

The following section describes the steps you need to perform to work with Qlik Replicate for Windows and Vertica as a target database in a Replicate task:

On the Qlik Replicate Server machine, install version 11.x of the following drivers:

- Vertica ODBC 64-bit
- VSQL CLI Client

Replicate server for Linux

The following section describes the steps you need to perform to work with Qlik Replicate for Linux and Vertica as a target database in a Replicate task:

1. On the Qlik Replicate Server machine, install Vertica ODBC driver 11.x for Linux.

Example:

```
vertica-client-11.1.0-0.x86_64
```

2. Makes sure that the `/etc/odbcinst.ini` file contains the following entry for Vertica, as in the following example:
[Vertica]
Driver = /opt/vertica/lib64/libverticaodbc.so
DriverODBCVer = 3.0
UsageCount = 1

Limitations

When replicating to a Vertica target, the following limitations apply:

- In "Batch optimized apply" Change Processing mode, binary data types (e.g. VARBINARY) are limited to 32500 bytes on the target.
- Replicating data with 4-byte emoji characters is not supported.

Security requirements

You must provide Vertica account access to the Qlik Replicate user. The Replicate user must also have the following privileges in the Vertica database:

- CREATE TABLE Privileges:
 - CREATE privilege on schema
- DROP TABLE Privileges:
 - USAGE privilege on the schema that contains the table or schema owner
- TRUNCATE Privileges (If the task is configured to truncate existing tables):
 - USAGE privilege on the schema that contains the table or schema owner

- ALTER TABLE (ADD/DROP/ RENAME/ALTER-TYPE COLUMN) Privileges:
 - USAGE privilege on the schema that contains the table or schema owner
- INSERT Privileges:
 - INSERT privilege on table
 - USAGE privilege on the schema that contains the table
- UPDATE Privileges:
 - UPDATE privilege on table
 - USAGE privilege on the schema that contains the table
 - SELECT privilege on the table when executing an UPDATE statement that references table column values in a WHERE or SET clause
- DELETE Privileges:
 - DELETE privilege on table
 - USAGE privilege on schema that contains the table
 - SELECT privilege on the table when executing a DELETE statement that references table column values in a WHERE or SET clause

Vertica target data types

The Vertica database for Qlik Replicate supports most Vertica data types. The following table shows the Vertica target data types that are supported when using Qlik Replicate and the default mapping from Qlik Replicate data types.



*Vertica does not support applying changes to binary data types in **Batch optimized apply** mode. For more information on **Batch optimized apply** mode, see *Change Processing Tuning (page 1181)*.*

For information on how to view the data type that is mapped from the source, see the section for the source database you are using.

For additional information about Qlik Replicate data types, see *Replicate data types (page 81)*.

Supported data types

Qlik Replicate Data Types	Vertica Data Types
BOOLEAN	BOOLEAN
BYTES	VARBINARY (Length)
DATE	DATE
TIME	TIME (p)
DATETIME	TIMESTAMP
INT1	INTEGER
INT2	INTEGER
INT4	INTEGER

Qlik Replicate Data Types	Vertica Data Types
INT8	INTEGER
NUMERIC	NUMERIC (p,s)
REAL4	FLOAT
REAL8	FLOAT
STRING	VARCHAR (Length multiplied by three) For example, STRING (50) becomes VARCHAR (150).
UINT1	INTEGER
UINT2	INTEGER
UINT4	INTEGER
UINT8	INTEGER
WSTRING	VARCHAR (Length in Bytes)
BLOB	VARBINARY (65,000)
CLOB	VARCHAR (65,000)
NCLOB	VARCHAR (65,000)

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add an Vertica target endpoint to Qlik Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button.
2. In the **Name** field, type a name for your database. This can be any name that will help to identify the database being used.
3. In the **Description** field, type a description that helps to identify the Vertica database. This is optional.
4. Select **Target** as the database **role**.
5. Select **Vertica** as the database **Type**.
6. In the **Server** field, enter the name of the Vertica server.
7. Optionally, change the default **Port** (5433).
8. Type the Vertica authentication information (**User Name**, **Password**) for the authorized user for this Vertica database. If you do not know this information, see your Vertica database Administrator (DBA).



- *This information is required. If you are using the **Advanced** tab to create a custom string, make sure to include the **User Name** and **Password** properties. See [Setting advanced connection properties \(page 1060\)](#) for more information.*
- *This information is case sensitive.*
- *If you want to set custom properties for this database, see [Setting advanced connection properties \(page 1060\)](#).*



Make sure that the Vertica user entered in the Vertica Authentication section has the correct access privileges. For information on how to provide the required privileges, see [Security requirements \(page 1057\)](#).]

9. In the **Database name** field, enter the name of the Vertica database.

Setting advanced connection properties

In the **Advanced** tab, you can set the following parameters:

- **Max file size:** Select or type the maximum size (in KB) of a CSV file before the file is loaded into the Vertica database. The default value is 32000 KB.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

11 Using the Qlik Replicate file channel

This topic describes how to use the Qlik Replicate File Channel as a source or target in a replication task.



If you use File Channel as both source and target, the version (build) of the Qlik Replicate servers on both sides should be identical.

In this section:

- [Setting Up Qlik Replicate File Channel tasks \(page 1063\)](#)
- [Working with the File Channel data files \(page 1065\)](#)
- [Prerequisites \(page 1062\)](#)
- [Security \(page 1062\)](#)
- [Limitations \(page 1062\)](#)
- [Using the File Channel as a source \(page 1067\)](#)
- [Using the File Channel as a target \(page 1069\)](#)

11.1 Prerequisites

- To work with the file-channel endpoint, you must install Qlik Replicate anywhere on the network for each LAN that you are working with.
- When using the File Transfer Service, make sure to open the firewall port defined in the File Transfer Service settings. For more information on working with the File Transfer Service, see the following sections:
 - *File Transfer Service (page 1269)*
 - *Setting advanced connection properties (page 1068)(File Channel source)*
 - *Setting advanced connection properties (page 1070) (File Channel target)*

11.2 Limitations

The following limitations apply:

- The File Channel endpoint does not support Full LOB mode.
- You cannot use the Full Load resume function if you are using the File Channel endpoint. To resume a Full Load operation, you must delete the original data and then run the task again.
- You must delete the File Channel folder before restarting an Apply Changes task.
- After modifying an existing transformation in a remote File Channel task, both the local and the remote File Channel tasks need to be *restarted* (by selecting the **Reload Target** run option in both tasks).
- Control tables defined for the local File Channel task but not for the remote File Channel task will not be created on the remote task's target endpoint.
For information on defining Control Tables, see *Control tables (page 1169)*.
- If the local task fails, or is stopped, it is possible that the Full Load operation in the remote task will not complete successfully.
- The source_lookup Data Enrichment function is not supported. For more information on this function, see *Data Enrichment functions (page 1157)*
- The [Unsuspend table option](#) is not supported.
- The following functionality is not supported by the local File Channel task and should be set in the remote File Channel task *only*:
 - Store Changes replication
 - Table-level or global transformations
 - The ignore DDL changes options (DROP, TRUNCATE, and ALTER) in the *Apply Changes Settings (page 1174)* tab
 - [Changing the schema](#) in which the Replicate Control tables will be created
 - [Stopping the Task after Full Load](#) options
- The following functionality is not supported by the remote File Channel task and should be set in the local File Channel task *only*:
 - [Metadata only](#) run options

11.3 Security

When using the File Transfer Service, file-channel files are always transferred over an encrypted session.

The session is encrypted as follows:

The client and server create an AES-256 session key using the Diffie-Hellman key exchange protocol (using the OpenSSL library). After the key is created, all file transfers between the client and the server will take place over a secure and encrypted communication channel.

However, even though the session is encrypted, communication between the client and the server may still be susceptible to man-in-the-middle attacks. A man-in-the-middle in possession of the session key would be able to intercept any data transferred between the client and the server.

To eliminate man-in-the-middle attacks, a "shared password" needs to be provided when configuring the local and remote file channel endpoints. Once the session is established, both the client and the server use the shared password to re-key the session key during the next packet exchange, thereby preventing the original session key from being used for man-in-the-middle attacks.

To sum up:

1. Strong encryption is used regardless of whether a password was provided.
2. Providing a password eliminates the risk of a man-in-the-middle attack.

For more information about the File Transfer Service, see *File Transfer Service (page 1269)*.

11.4 Setting Up Qlik Replicate File Channel tasks

To replicate data using the file channel, you must set up two tasks of the following type:

- *Local task (page 1063)*
- *Remote task (page 1063)*



When using file channel, Change Tables can be enabled for the remote task but not for the local task (enabling Change Tables for the local task will result in remote task failure).

Local task

You set up the local task using the File-Channel endpoint as a target. The binary file created in this task is used as the source for one or more remote tasks using the File-Channel source endpoint.

The local task replicates data from a Qlik Replicate supported endpoint to the file channel. If you changed the default folder for storing data files (during the installation), then you must specify the location of the binary file created by the file channel. This location can be anywhere in your system. For more information on setting up a local task, see *Using the File Channel as a target (page 1069)*.

Remote task

Remote tasks use the File Channel as a source endpoint. You use the file created by the local task for this source. You can replicate the data to any endpoint that is supported by Qlik Replicate. You define the location of the File-Channel file as the remote location where the file was created. The data is pushed over the network to the defined location anywhere in your system. You can also define more than one location for the replicated data. In this case, define a separate remote task for each location.



For remote tasks configured with the File Channel source endpoint, you must enable Full Load replication and set the **If target table already exists** option in the task settings' Full Load Settings (page 1171) tab to **Do Nothing**.



The [attrep_cdc_partitions](#) Control Table is not supported when using File Channel as a source endpoint.

If you want to push the data to an endpoint that is not in your LAN, use the *File Transfer Service* (page 1269) to send the files created in the local task to the remote location.

When you run the remote task, data is sent to the target in the following instances:

- The first time you run the task as a full load.
- Each time changes are made to the file. In this case, change processing takes place.

When the remote task runs, it will continuously look for the source file until the task is stopped. When the file is found, the data is replicated to the target endpoint. If no source file is found, an error is displayed; however, the task will continue to check for the correct file. Therefore, it is recommended that you run the local task first to ensure that the file exists.



To replicate tables that were added to the local file channel task after the initial full load, you need to reload both the local and the remote file channel tasks.

For more information on setting up a remote task, see *Using the File Channel as a source* (page 1067).

Replicating to multiple targets (distribution)

You can use the File Channel to distribute from a single source endpoint to multiple targets, either of the same type (e.g. Microsoft SQL Server to Microsoft SQL Server) or of different types (e.g. Microsoft SQL Server to Oracle and SAP Sybase ASE).

To do this:

1. For each of the target endpoints, define a separate (remote) task that replicates from the File Channel source to the target endpoint. In the **Advanced** tab of the File Channel source settings, make sure to clear the **Delete processed files** check box. This ensures that the File Channel files will be available for distribution as required.
2. Define a local task that replicates from the source endpoint to a File Channel target.
3. Run the local task (this will create the File Channel files required by the remote task).
4. For each of the remote tasks, select which tables to replicate (from the File Channel source) and optionally apply Filters and Transformations to them.
5. Run the remote tasks.

For more information on defining tasks, see *Defining and managing tasks* (page 109).

For information on Filters and Transformations, see *Customizing tasks* (page 1082).



By default, all the metadata for the selected source tables is replicated from the Local task (page 1063) to the Remote task (page 1063). This allows you to remove, add and transform tables in the remote task as needed. However, if you want the tables in the source and target endpoints to be identical, you can prevent replication of the metadata (and thereby shorten the processing time) by specifying `providerremotemetadata=N` in the **Override connection string parameters** field of the File Channel target's **Advanced** tab.

Adding tables to a running remote task

When distributing to multiple targets, it is possible to replicate a different subset of tables to each target if necessary. Before starting the task, you can select which tables to replicate using the standard procedure described in *Selecting tables and/or views for replication* (page 118). However, if the task is already running, you need to perform the following procedure:

1. Stop the remote task.
2. Add the desired tables (as described in *Selecting tables and/or views for replication* (page 118)).
3. Resume the remote task. The newly added tables will be marked as “Queued”.
4. Reload the newly added tables in the local task (by selecting the tables and clicking the **Reload** icon in Monitor view).

For information on removing specific tables from a replication task, see *Removing specific tables/views from a replication task* (page 122).



Adding tables to the remote task is not supported in Apply Changes (CDC-only) replication tasks. For more information on the available replication options, see *Adding tasks* (page 110).

11.5 Working with the File Channel data files

The File Channel stream data files are encoded in an internal binary format. For full-load operations, the File Channel binary files contain packed data records for each of the table records and an end-of-file (EOF) record. For change-processing operations, the file contains:

- A packed data record for each DDL and/or DML change.
- A **begin-load-table** record with the stream name that marks the beginning of table loading.
- A packed table-definition record with the table metadata. These records come before each DDL and **begin-load-table** record.

You do not need to work directly with the file-channel files, however if you find it necessary to work with them they are located in the File-Channel Directory Structure described below.

File-Channel directory structure

The file-channel directory contains the following files and folders:

- **s_msgs**: This folder contains messages sent from the source side to the replication server on the remote target side.
Messages are removed from this folder at the source side when an acknowledgment message is received stating that the file was transferred successfully or possibly with a timeout.
Messages are removed from this folder at the target side after they are read.
This folder contains the following files:
 - **s_msgs/xxxxxxxx.fcm**: This file contains a JSON message from the source side to the target side.
 - **yyymddhhMMss.mtd**: This file contains the captured tables list.
- **s_status**: This folder contains status updates from the source side to the target side. Status updates appear as a fixed name file that is periodically updated. This file lists the last processed target status file. It receives the [t_status/ccccccc.fcs](#) file. These files are deleted when the file-channel source endpoint finishes reading the file. You can configure the file-channel source to keep the files, if necessary. See *Setting advanced connection properties (page 1068)* for more information.
- **t_status**: This folder contains status updates from the target side to the source side. Status updates appear as an infinite set of data files that are created according to a specific schedule. These files are sent from the target by the source. The folder contains also a fixed name file that is updated with the last created status file name. It contains the following file:
 - **t_status/ccccccc.fcs**: This is a file channel status file (.fcs) where the file name is a hexadecimal counter of length 8. These files will be transferred in order with the lower numbers transferred first. If you need to view them, you should order them by timestamp because alphabetical ordering will not be consistent with the hexadecimal name.
File channel status files are deleted by the source after being read and by the target when source status file indicates that this file was already processed.
You can configure the maximum amount of time that the files are kept before a new file is created as well as the maximum file size for each file. The minimum file size is 50 MB.
For more information, see *Setting advanced connection properties (page 1070)*.
- **streams/<stream-name>**: This folder contains stream sub-folder, one sub-folder per stream. A stream represents a finite or infinite set of data files being sent from the source to the target. The file channel allows creating and destroying named streams dynamically. For example, there can be a fixed-named stream `cdc` (`streams/cdc`) and there could be a dynamically created stream `loadXXXXXXXX` that can be removed at the source side when a status update from the target is received (for example, when processing completed) in the **t_status** folder.
You can configure the maximum number of streams and the maximum disc space for each stream. For more information, see *Change Processing (page 1174)*.
This folder contains the following file:
 - **streams/<stream-name>/ccccccc.fcd**: This is a file channel data file (.fcd) where the file name is a hexadecimal counter of length 8. These files are processed at the target in order or in parallel depending on the case. However, the files are transferred in order with the lower numbers transferred first.
File channel data files are deleted by the source when transferred successfully and by the target when processed.
You can configure the maximum amount of time that the files are kept before being creating a new file and the maximum file size for each file. The minimum file size is 10 MB and the minimum time that a file is kept is 5 seconds.

11.6 Using the File Channel as a source

The File Channel source endpoint is a Qlik Replicate endpoint that consumes and applies the contents of a file channel directory structure that was produced by a corresponding File Channel target endpoint.

11.7 Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.

To add the File Channel source to Qlik Replicate:

1. In **Tasks** view, click **Manage Endpoint Connections** toolbar button to open the **Manage Endpoint Connections** window.
2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the endpoint being used.
3. In the **Description** field, type a description that helps to identify the information being replicated to the file. This is optional.
4. Select **Source** as the endpoint **Role**.
5. Select **File Channel** as the endpoint **Type**.
6. Type the full path to the **Storage Folder** where the File Channel files will be created. The default path when *not* using the File Transfer Service is:

`C:\Program Files\Qlik\Replicate\data\tasks\<<task_name>`

If you are using the File Transfer Service, the default path is:

`C:\Program Files\Qlik\Replicate\data\endpoints\<<file-channel_db_name>\fc`



The Replicate File Transfer Service always transfers the local file channel task's files to the default directory on the remote system (`C:\Program Files\Attunity\Replicate\data\endpoints\<<remote_file-channel_db_name>\fc`). Consequently, if you are using the File Transfer Service, ensure that the default directory always has enough space for the incoming files. For more information on using the File Transfer Service, see [File Transfer Service \(page 1269\)](#) and [Setting advanced properties](#).



*The actual size of the File Channel files is usually three-to-four times larger than the size of the source data. You should therefore make sure that the location of the specified **Storage Folder** has sufficient disk space.*

This folder should be in a location that is accessible from anywhere in the WAN you are working with.



To determine if you are connected to the endpoint you want to use or if the connection information you entered is correct, click **Test Connection**.
If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box.
To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the test connection fails.

7. Click **OK** to finish the setup and save the changes.

11.8 Setting advanced connection properties

You can set the following properties in the **Advanced** tab:

- **Input files are received via file transfer service:** Select this check box to receive the source input files using the Replicate File Transfer Service.
 - **Password:** The password that will be used to establish a secure connection with the File Channel Target.



When using the File Transfer Service, an agreed upon password is required in order to establish a secure connection between the File Channel Source and the File Channel Target. Accordingly, the password specified in the File Channel Source settings and the password specified in the File Channel Target settings must be identical.

For more information about the File Transfer Service, see *File Transfer Service (page 1269)*.

- **Delete processed files:** Select this check box to delete the File Channel files after the data has been replicated to the target endpoint.
You should clear this check box if other tasks need to use the files.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

11.9 Using the File Channel as a target

The File-Channel target endpoint is a Qlik Replicate endpoint that creates and maintains a file-based directory structure containing replication artifacts (task definitions, metadata, full load data, CDC data and status updates). This file channel directory structure is consumed by a corresponding File-Channel source endpoint in a different task and possibly in a remote location.

Setting general connection properties

This section describes how to configure general connection properties. For an explanation of how to configure advanced connection properties, see **Setting advanced connection properties**.



- The **Type** is different depending on the type of file you are creating, however the information you enter is the same for all file types.
- All files are used as targets, however you can use a Qlik Replicate file as a source only after you created the file by loading data into it as a target.

To add the File Channel target to Qlik Replicate:

1. In the Qlik Replicate Console, click the **Manage Endpoint Connections** toolbar button to open the **Manage Endpoints Connections** dialog box. Then click the **New Endpoint Connection** button.
2. In the **Name** field, type a name for your endpoint. This can be any name that will help to identify the endpoint being used.
3. In the **Description** field, type a description that helps to identify the information being replicated to the file. This is optional.
4. Select **Target** as the endpoint **role**.
5. Select **File Channel** as the endpoint **Type**.
6. If you changed the default data folder during installation, type the full path to the **Storage Folder** (e.g. D:\data\tasks\) where the file is being created. Otherwise, you can leave this field empty. Note that this field will be ignored when the **Transfer files to remote file channel** option is enabled in the **Advanced** tab.



*To determine if you are connected to the endpoint you want to use or if the connection information you entered is correct, click **Test Connection**. If the connection is successful a message in green is displayed. If the connection fails, an error message is displayed at the bottom of the dialog box. To view the log entry if the connection fails, click **View Log**. The server log is displayed with the information for the connection failure. Note that this button is not available unless the*



test connection fails.

7. Click **OK** to finish the setup and save the changes.

Setting advanced connection properties

You can set the following properties in the **Advanced** tab:

- **Max file size (KB):** Click the arrows to select, or type the maximum file size (in kilobytes) allowed for the files created in the target.
- **Limit storage size to (MB):** To allocate a specific amount of disk space to the File Channel files, enable this option and then specify the amount of disk space to set aside (using the arrows or by typing). When the limit is reached, Qlik Replicate will stop writing the files to the designated storage.
- **Max batching time interval (seconds):** Click the arrows to select, or type the maximum time (in seconds) for files to be batched before being written in a single operation.
- **Transfer files to remote file channel:** Select this check box to transfer files to the File Channel Source (on the remote Qlik Replicate Server) using the Qlik Replicate File Transfer Service. This can dramatically improve transfer speeds when the source endpoint and the target endpoint are located on different LANs. For more information about the Qlik Replicate File Transfer Service, see *File Transfer Service* (page 1269).
 - **Remote file transfer service host:** The host name or IP address of the computer on which the Qlik Replicate File Transfer Service is running.
 - **Remote file transfer service port:** The port on the remote computer through which the files will be transferred (from the storage folder to the remote file channel).
 - **Remote file transfer service endpoint name:** The name of the File Channel Source endpoint on the remote machine.
 - **Additional remote file channels:** When sending to multiple File Channel Source endpoints, specify the target destinations using the following format:
`file_channel_db_name@host:port,file_channel_db_name@host:port`
 - **Max transfer streams:** The maximum number of streams to use when transferring the files. Adjust the number of streams as required to optimize transfer speeds.
 - **Password:** The password that will be used to establish a secure connection with the File Channel Source.



When using the File Transfer Service, an agreed upon password is required in order to establish a secure connection between the File Channel Source and the File Channel Target. Accordingly, the password specified in the File Channel Target settings and the password specified in the File Channel Source(s) settings must be identical.

Internal parameters

Internal parameters are parameters that are not exposed in the UI. You should only use them if instructed by Qlik Support.

To add internal Qlik Replicate parameters:

1. Click the **Internal Parameters** link.
The **Internal Parameters** dialog box opens.
2. In the edit box, type the name of the parameter you need to add and then click it.
3. The parameter is added to the table below the search box with its default value.
4. Change the default value as required.
5. To reset the parameter value to its default, click the "Restore default value" icon at the end of the row.

Settings summary

You can view a summary of your settings by clicking the **Setting Summary** link. This is useful if you need to send a summary of your settings to Qlik Support.

12 Using the Log Stream

This section explains how to use the Log Stream feature. Log Stream enables a dedicated Replicate task to save data changes from the transaction log of a single source database and apply them to multiple targets, thereby eliminating the overhead of reading the logs for each target separately.

In this section:

- *Terminology (page 1073)*
- *Overview (page 1073)*
- *Step 1: Create a Log Stream target endpoint (page 1074)*
- *Step 2: Create and run a Log Stream Staging task (page 1075)*
- *Step 3: Create a replication task to transfer the staged data to the target (page 1078)*

12.1 Terminology

The following terms are used throughout this section.

- **Log Stream** - The name of the functionality that allows users to stream and store transaction log data as files on a disk.
- **Log Stream target endpoint** - The name of the component that writes the streamed log data to the Log Stream Staging folder.
- **Log Stream Staging folder** - The location of the files created from the streamed log data.
- **Log Stream Staging task** - The name of the task type that writes the data to the Log Stream Staging folder.
- **Replication task** - The name of the task type that reads the data from the Log Stream Staging folder and writes it to the final target endpoint.

12.2 Overview

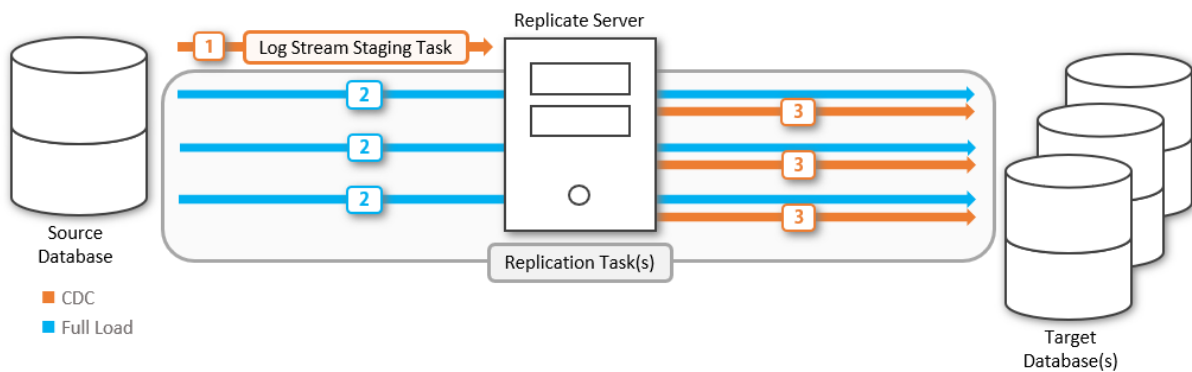
Log Stream enables a dedicated Replicate task to save data changes from the transaction log of a single source database and apply them to multiple targets, thereby eliminating the overhead of reading the logs for each target separately. Data changes from the log are written to a file in the Log Stream Staging folder enabling one or more targets to access them via separate Replication tasks.

Log Stream provides the following enhancements:

- When multiple CDC tasks read changes from large databases that share a single transaction log (e.g. Oracle and IBM DB2 for z/OS), the log is polled for only one target (the Log Stream target) instead of being polled separately for each target. This improves performance while greatly reducing the load on the source database and network.
- Changes are continuously written to the log stream regardless of the availability of the target endpoint. This is especially beneficial in situations where the original source transaction logs are no longer online or cannot be accessed efficiently. Additionally, this approach significantly reduces latency when the target endpoint becomes available again as the logs are read from the staging folder instead of from the source database.
- As the changes are constantly streamed to the Replicate server machine, the Replicate SLA that requires an extended log retention policy to be defined on the source in case the target cannot be accessed is no longer applicable. This allows you to reallocate valuable resources on the source while ensuring that log files will be up-to-date (on the Replicate Server machine) even if the target become unavailable.
- When replicating a single source database to multiple targets, each target can have its own subset of the log stream data (tables, rows, etc.) and be started, stopped, and scheduled independently of the others.

The flow

As mentioned above, a Log Stream setup provides benefits both when replicating to a single target and when replicating to multiple targets. The following diagram illustrates the end-to-end flow of a Log Stream setup with multiple target endpoints:



1. A Log Stream Staging task streams the source logs from the specified source endpoint to a designated staging folder on Replicate Server (specified in the Log Stream target endpoint settings).

Then, at least one Replication task:

2. Performs a Full Load of the source data to the designated target endpoint(s).
3. Transfers the changes from the Log Stream Staging folder to the final target endpoint(s).



The Log Stream Staging task and replication task must both run on the same Replicate server.

12.3 Step 1: Create a Log Stream target endpoint

To create a new target endpoint:

1. In the **Manage Endpoint Connections** window, select **New Endpoint Connection**. For information on how to add an endpoint, edit and view its configuration, see *Defining and managing endpoints* (page 114).
2. In the **General** tab, specify the following information:
 - Name for the endpoint. For example, `Log Stream Staging Target`.
 - Optionally, provide a description.
 - Select the **Target** role option.
 - From the **Type** drop-down list, select **Log Stream**.
 - In the **Storage path** field, specify the full path in which to store the streamed data (i.e. the binary file with the record changes). You cannot specify a relative path.



Make sure the staging folder is located on the Replicate Server machine. Qlik recommends using a block device (physical, virtual or iSCSI-based) for storing the streamed data. Using NFS or SMB-based storage is not supported due to the associated latency which could greatly degrade the data transfer performance, as well as due to reduced reliability and compatibility issues.

- **Batch:** Specify when to close the batch of operations that will be applied to the target endpoint:
 - **Apply batch after (seconds):** Length of time that the batch should run. Default is 5 seconds. Maximum permitted time is 60 seconds.
This parameter effects the start time of the Replication task; until at least one batch is written by the Log Stream Staging task, the Replication task will not start.
 - **Apply batch when number of changes reaches:** Maximum number of change operations to include in the batch. Default is 10,000. Maximum permitted number of operations is 1,000,000.
 - **Retention:** Specify when the staging file should be deleted (note that active files will not be deleted):
 - **Delete staging files after (hours):** Select the check box and specify the maximum time before a file is deleted. Default is 48 hours. Maximum permitted time is 10,000 hours.
 - **Delete oldest files when the total size of all staging files exceeds (MB):** The maximum size that you want to allocate for the staging folder. If the specified size is reached, Replicate will start deleting files from the oldest to the newest until the total size falls below the upper limit.
Default is 100,000 MB. The minimum size should be at least twice the defined rollover size. Maximum permitted size is 1000 TB.
 - **Rollover:** Specify when to start writing to a new staging file:
 - **Roll over file after (minutes):** Number of minutes after which a new staging file should be started. Default is 120 minutes. Maximum permitted time is 10,080 minutes (one week).
 - **Roll over files larger than (MB):** Size of the file after which a new staging file should be started. Default is 500 MB. Maximum permitted size is 100,000 MB.
 - **Compression:** Optionally move the slider to change the file compression level. The higher the compression level, the greater the processing latency will be. Therefore, a lower compression level (e.g. 2) is recommended when processing speed is more important than saving disk space.
3. Click **Test Connection** to verify the information you entered and the availability of the target database.
 4. Click **Save**.

12.4 Step 2: Create and run a Log Stream Staging task

Once the Log Stream target endpoint is created, a Log Stream Staging task must be defined that will replicate any changes to the original source endpoint data and metadata to the Log Stream Staging folder.

The Log Stream Staging task creates a “Staging” file in the Log Stream Staging folder. This file contains the changes from the source database transaction log as a compressed binary file.

Before running the Log Stream Staging task, you must define and add the source endpoint where your data is currently stored and the target endpoint where you want to replicate the data – in this case, the Log Stream target endpoint. After you add the endpoints to your task, you can select the tables from the source endpoint that you want to replicate. For information on selecting tables, see *Selecting tables and/or views for replication (page 118)*.



- *Log Stream Staging tasks can be imported or exported. For details, see [Exporting and Importing Tasks](#).*
- *Log Stream Staging tasks can process files created by previous versions of Qlik Replicate. The files will include an indication of the version number.*

To define a Log Stream Staging task:

1. In the Tasks view, click New Task. The **New Task** dialog box opens.
2. Specify a name for the task.
3. Choose the **Log Stream Staging** replication profile option.

The screenshot shows the 'New Task' dialog box with the following details:

- Name:** New Task 11
- Description:** (Empty field)
- Replication Profile:** Select the profile that best describes your replication use case. Options: Unidirectional, Bidirectional, **Log Stream Staging** (selected).
- Task Options:** Select the replication options for this task. Option: **Stage Changes** (selected).
- Task Options Description:** Changes read from the source are staged in a local log stream for consumption by other replication tasks.
- Buttons:** OK, Cancel

4. Click **OK**.

- The next step in the Log Stream Staging task is to select the source endpoint where your data is currently stored, and the Log Stream target endpoint.



- All source endpoint connections are available for selection.
- Only target endpoints of type "Log Stream" are available for selection.
- Log Stream target endpoints that are already associated with other Log Stream Staging tasks are not available for selection.

The following figure provides an example of selected source and target endpoints for a Log Stream Staging task:

Log Stream Staging Task



- In the right pane in the Console, click **Table Selection** to select the source tables that you want to replicate.
The task will create the metadata in the Log Stream staging folder and then capture the changes from the specified source endpoint.
The task is now ready to run.
- To execute the Log Stream Staging Task, click the **Run** button.



The Log Stream Staging task must be run before the Replication task described in Step 3: Create a replication task to transfer the staged data to the target (page 1078).

You can view and monitor the results of the replication in real-time.

Log Stream Staging limitations and considerations

- The **Full Load** and **Store Changes** replication options are not relevant.
- In the **Advanced Run Options** dialog box, only the following options are supported:
 - **Tables are already loaded. Start processing changes from: Date and Time.**
 - Recovery: **Recover using locally stored checkpoint**
- In the **Task Settings** window, under the **Target Metadata** tab:
 - **Target table schema** is unavailable
 - The **Allow unlimited LOB size** option is disabled.
 - **Limit LOB size to (KB)**: Default is 8 KB; maximum permitted size is 102,400 KB
- In the **Task Settings** window, the following tabs and/or settings are not available:
 - **Metadata > Control Tables** tab
 - **Full Load** tab
 - Under the **Change Processing** tab: **Apply Changes Settings** tab, and **Store Changes Settings** tab.
 - In the **Change Processing Tuning** tab: **Batch Tuning** and **Miscellaneous Tuning** settings
 - **Change processing mode**: only **Transactional apply** is available
 - Error handling: **Data Errors** tab, **Table Errors** tab, **Apply Conflicts** tab
 - **Character Substitution** tab
- In the **Table Settings** window:
 - The following tabs are not available: **Transform**, **Filter**, **Parallel Load**, and **LOB Column Handling**
 - In the **General** tab, the **Map to target table** section is unavailable.
- Global Transformations are not available.
- The source_lookup Data Enrichment function is not supported. For more information on this function, see *Data Enrichment functions (page 1157)*
- Source endpoints that do not support CDC (such as, ODBC, Hadoop) cannot be used as sources for Log Stream Staging tasks.
- File Channel cannot be used as a source for a Log Stream Staging task.
- When the source endpoint is IBM DB2 for z/OS or IBM DB2 for iSeries, the option [in the Tables settings'](#) **Transform tab** for changing the character set or data type for a specific input column is not supported.

12.5 Step 3: Create a replication task to transfer the staged data to the target

After the data is updated in the Log Stream Staging folder, you then need to create a Replication task to transfer the changes from the Log Stream Staging folder to the final target endpoint.



If you stop the task and define a metadata transformation for one of the tables (such as dropping a column), make sure the **DROP and CREATE table** option is selected (the default) in the Task Settings' **Full Load Settings** tab before resuming the task.

To do this, you must first duplicate of the source endpoint that has the data you want to replicate. Then, you can create a Replication task in which the source endpoint is the duplicated source endpoint of the Log Stream Staging task and the target is any required target endpoint.

Duplicate the source endpoint and add it to the log stream staging task



- When duplicating a source endpoint for use in a Replication task, you can define credentials that are different to those used in the Log Stream Staging task.
- Instead of duplicating the source endpoint, you can create a new endpoint and give it the same properties as the existing one.

To duplicate a specific endpoint and connect it to the Log Stream Staging task:

1. In the left panel of the **Manage Endpoint Connections** window, select the original source endpoint defined for the Log Stream Staging task.
2. Click the **Duplicate** button.
3. In the **General** tab, do the following:
 1. Replace the name of the source endpoint prefixed by "Copy of ..." with a meaningful name for the duplicated endpoint (for example, Oracle1 with Log Stream).
 2. Select the check box **Read changes from log stream staging folder**.



This check box only appears if the duplicated source is connected to the Log Stream Staging endpoint.

3. From the **Log stream staging task** drop-down list, select the name of the relevant Log Stream Staging task.



All Log Stream Staging tasks that are related to the duplicated source are provided in the drop-down list. If a task was never run, "Not Run" is displayed next to its name.

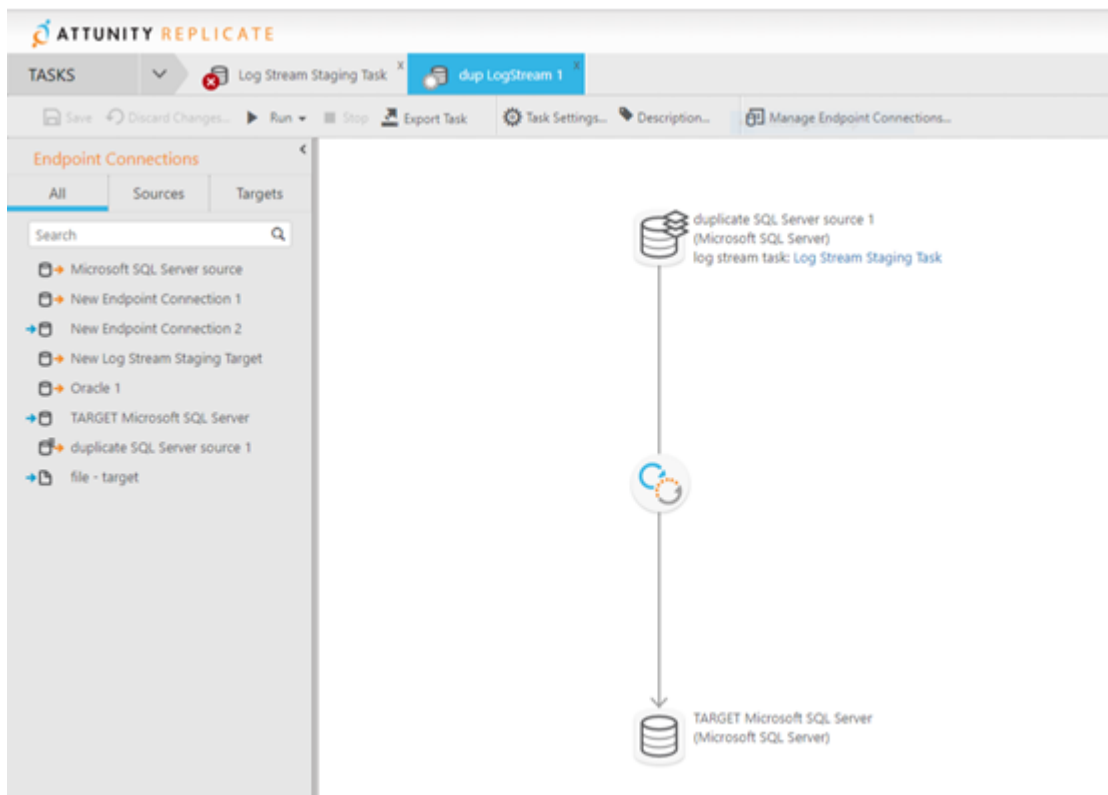
4. Click **Test Connection** to determine if you are connected to the Log Stream staging task and if the information you entered is correct.
5. Click **Save** and then **Close**.

Create the replication task using the duplicated source endpoint

Before you begin: Make sure that the Log Streaming Staging task is running and that it includes all the tables you need for the Replication task.

To create a new Replication task using the duplicated source endpoint

1. In the **Tasks** view, click **New Task**. The New Task dialog box opens.
2. Specify a meaningful name for the replication task.
3. Choose the **Unidirectional** replication profile option.
4. If required, select the **Full Load** or **Store Changes** task options.
5. Click **Apply Changes**, and then **OK**.
6. Select the required source with Log Stream endpoint from the list of endpoint connections, and drag it to the top circle in the endpoints map.

Example:

- A Log Stream indication is added to the source endpoint icon.
- The Task Name links directly to the Log Stream Staging task.

7. Select the required target endpoint from the list of endpoint connections, and drag it to the bottom circle in the task map.
8. From the right pane in the Console, click **Table Selection** and select the tables that you want to replicate. You can select any of the tables that were included in the Log Stream Staging task.
9. Click the **Run** button to execute the Replication task.
The Replication task will load the tables to the final target endpoint, replicating the full load from the original source endpoint and the changes from the Log Stream Staging folder.

Replication task limitations and considerations

- The Bidirectional replication profile is not supported.
- In the **Target Metadata** tab of the **Task Settings** dialog box, if you want to enable the **Replicate LOB columns** option, the **Limit LOB size to** value must be the same as the value specified in the Log Stream Staging task. Note however that if the **Replicate LOB columns** option is disabled for the Log Stream Staging task, it cannot be enabled in the Replication task.
- In the **Table Settings** window, most tabs are available and functional with the exception of the **LOB Column Handling** tab which is not available (at table level).
- The source_lookup Data Enrichment function is not supported. For more information on this function, see *Data Enrichment functions (page 1157)*
- The **Source change position (e.g. SCN or LSN)** [Advanced Run option](#) is only supported with the Oracle source endpoint.
- When the [Use all table partitions](#) Parallel Load method is enabled, source data in a partition that was created *after* the Log Stream Staging task started, will not be replicated to the target defined for the Replication task.

13 Customizing tasks

This section describes how to customize a replication task. For example, you can create new tables or columns for the target endpoint or select only some of the data from each column to be replicated. This is done using transformations and filters.



Although the descriptions in this section only refer to tables, the procedures described herein are applicable to views as well. When a transformation is defined for a view, the word "View(s)" appears in the UI instead of the word "Table(s)".

In this section:

- *Table Settings (page 1082)*
- *Defining global rules (page 1107)*
- *Using the Expression Builder (page 1137)*
- *Task Settings (page 1165)*

13.1 Table Settings

In the **<Table_Name> - Table Settings** window, you can define how the data for each individual table/view is replicated to the target.



Some of the table settings are not available in a Log Stream Staging setup.

For information on the availability of table settings in a Log Stream Staging setup, see *Using the Log Stream (page 1072)*.

To open the Table Settings window:

1. Open the task you are working with.
For information on opening a task, see *Editing a replication task (page 125)*.
2. In Designer view, select the desired table from one of the following tabs on the right of the console:
 - The **Patterns and Selected Tables** tab - if the desired table was explicitly selected.
 - The **Full Table List** tab - if the desired table was selected using a table inclusion pattern.
For information on how to define table selection patterns, see *Creating table/view selection patterns (page 122)*.
3. Click the **Table Settings** button above the table list.
The **<Table_Name> - Table Settings** window opens.
4. In the **Table Settings** window, perform any of the following tasks:
 - *Performing General tasks for a single table/view (page 1083)*
 - *Defining transformations for a single table/view (page 1084)*
 - *Using filters (page 1092)*

- *Parallel Load (page 1098)*
 - *Handling LOB columns (page 1103)*
 - *Message format (page 1106)*
 - *Full Load (page 1106)*
5. Click **OK** to close the **Table Settings** window.
 6. Click **Save** in the main toolbar to preserve the table and column information for this task.

To restore the default table values:

- Click **Restore Table Defaults** at the bottom left of the **Table Settings** window. This option is available in all tabs.
Any changes you made will be discarded and the table's default settings will be restored.



*The names of modified tables will be followed by the word **(changed)**, enabling you to easily identify which tables have been modified.*

Performing General tasks for a single table/view



Although the descriptions in this section only refer to tables, the procedures describe herein are applicable to views as well. When a task is being performed for a view, the word "View(s)" will appear in the UI instead of the word "Table(s)"

The **General** tab in the Table Settings window displays basic information about the selected table and allows you to define new names for the table/schema on the target as well as override the default tablespace for the table and its index (Oracle target only).

To edit the general table settings:

1. Open the *Table Settings (page 1082)* window.
2. Click the **General** tab on the left side of the window, as shown below.

DEMO.EMPLOYEE - Table Settings

General

Transform

Filter

Parallel Load

LOB Column Handling

Source Table Details

Table Schema	DEMO
Table Name	EMPLOYEE
Record identifier	Non clustered primary key
Identifier key/index name	_SYS_TREE_RS_#8726962_#0_#P0
Table tablespace	
Index tablespace	
Approximate record size (Byte)	79
Estimated record count	15
Estimated table size (MB)	0
Load Order	Normal Priority

Map to target table

Table Schema:

Table Name:

Table tablespace:

Index tablespace:

Restore Table Defaults

OK Cancel

In the **Map to target table** section, the following options are available:

- **Table Schema:** Specify the schema in which you want the table to be created on the target.
- **Table Name:** Specify a new name for the table on the target.
- **Table tablespace:** This option is only available when the task is defined with an Oracle target endpoint.
Specify the name of the tablespace in which you want the table to be created on the target. By default (i.e. when this field is empty), the table will either be created in the source table tablespace on the target (when replicating from an Oracle source) or in the default tablespace (when replicating from any other source).
- **Index tablespace:** This option is only available when the task is defined with an Oracle target endpoint.
Specify the name of the tablespace in which you want the table's index to be created on the target. By default (i.e. when this field is empty), the index will either be created in the source table tablespace on the target (when replicating from an Oracle source) or in the default tablespace (when replicating from any other source).

Defining transformations for a single table/view



Although the descriptions in this section only refer to tables, the procedures describe herein are applicable to views as well. When a transformation is defined for a view, the word "View(s)" will appear in the UI instead of the word "Table(s)".

This section describes how to define data transformations. Data transformations are performed when the task is run. They are optional. If you do not define any transformations, the data is replicated "as is" from the source to the target.

Qlik Replicate lets you make the following changes to the tables and columns:

- Rename any column for the target table
- Delete a target column
- Change the data type and/or the length of any target column
- Add additional target columns
- Designate which target columns (i.e. segments) will comprise the Unique Index
- Recalculate the data

Limitations

Transformations are subject to the following limitations:

- Calculating columns of right-to-left languages is not supported.
- Transformations cannot be performed on columns that contain special characters (e.g. #, \, /, -) in their name.
- Transformations cannot be performed on columns that have a pound character (#) in their name.
- The only supported transformation for LOB/CLOB data types is to drop the column on the target.
- Using a transformation to rename a column and then add a new column with the same name is not supported.

You can use the method described here for transformations that are specific to a single table or a few tables in your task. To make a similar change over multiple tables, see *Starting the Global Transformation Rules wizard (page 1107)*.

For an explanation of how to configure transformations, see *Using the Transform tab (page 1086)*.

To define a data transformation for a single table:

1. Select the table you want to transform and open the *Table Settings (page 1082)* window.
2. Click **Transform** on the left side of the window.

The following figure shows the information in the **Transform** tab of the **Table Settings** window.

HR.DEPARTMENTS Table Settings ×

General **Transform** Filter

Source table: HR.DEPARTMENTS By default include all columns

Input Add Column

Key	Name	Type	Expression
	DEPARTMENT_ID	INT2	
	DEPARTMENT_NAME	STRING(30)	
	MANAGER_ID	INT4	
	LOCATION_ID	INT2	

Restore Table Defaults OK Cancel

Using the Transform tab

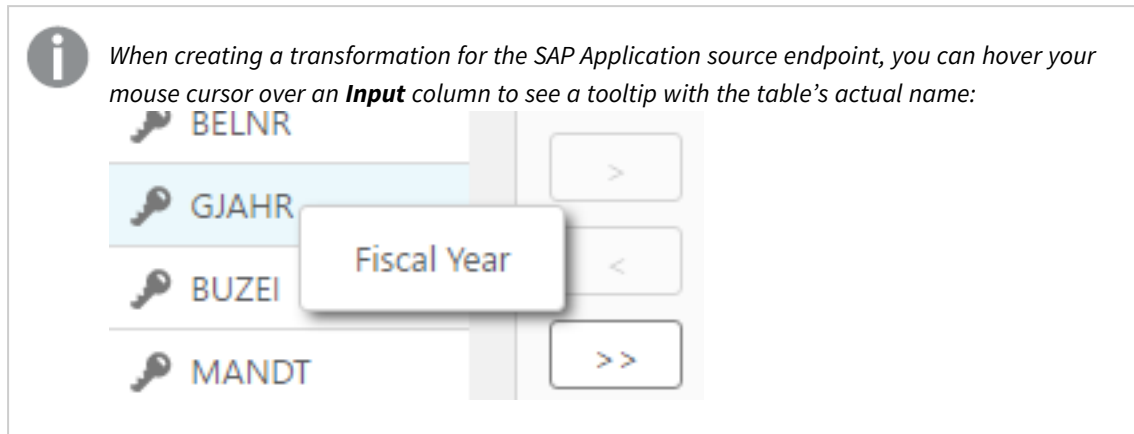
In the **Transform** tab, you can define transformations using Replicate's built-in functionality.



Customers that requires functionality not provided by Replicate's built-in transformations can write their own transformations, and then access them from the [Replicate Expression Builder](#). For an explanation of how to create user-defined transformations (requires basic programming skills), see [User-defined transformations](#) (page 1163).

The **Transform** tab in the **Table Settings** window consists of the following elements:

- **Input:** This lists the columns on which you can perform transformations.



- **Output:** This table shows the defined output for the columns in the table where you are performing the transformation(s). See Transformation Options below for information on how to change the default output.

Limitations and considerations


- Dropping a column, saving your changes, and then adding a column with the same name and defining an expression corresponding to the dropped column's data, is not supported. If you mistakenly drop a column, simply add the column back again without an expression.
- If you stop a task and define a metadata transformation for one of the tables (such as dropping a column), make sure the **DROP and CREATE table** option is selected (the default) in the Task Settings' **Full Load Settings** tab before resuming the task.
- In homogeneous replication tasks (such as Oracle to Oracle), modifying a single table column (by changing the column data type or length for example), will break the homogeneity for the entire table.





Transformation options




The following table describes the transformation options available in the **Transform** tab.

Transform actions

To	Do This
Rename a column	<p>Select the Name column for the table column you want to change. Type in a new name.</p> <p>The top right corner turns blue when the name is changed. To view the original name, hover the mouse pointer over the field and the original name is displayed.</p>
Set a column as a primary key/unique key or disable a column's primary key/unique key	<ol style="list-style-type: none"> 1. Select the desired row in the Output table and then click the cell in the Key column. A key icon will be displayed. 2. Repeat to set primary keys/unique keys for additional columns. 3. To disable the primary key/unique key, click the key icon.

To	Do This
Change the data type for a column	<p>Select the Type column for the table column you want to change and select a new data type from the drop-down list. Make sure that the data type you select is compatible with the data in that column.</p> <p>For a description of Qlik Replicate data types, see <i>Replicate data types (page 81)</i>.</p> <p>For information about data-type mapping from the native endpoint to Qlik Replicate data types, see the chapter for the endpoint you are using. For a list of supported databases, see <i>Supported Platforms and Endpoints (page 1313)</i>.</p>
Change the data subtype for a column	<p>This option is available for the CLOB, NCLOB, STRING, and WSTRING data types only.</p> <p>Select the Subtype column for the table column whose data type you want to change, and then select either JSON or XML from the drop-down list. Make sure that the data in the column is compatible with the selected subtype. The default is Regular, which means that data type in the Type column will be used with no subtype.</p> <p>For a description of Qlik Replicate data types, see <i>Replicate data types (page 81)</i>.</p> <p>For information about data-type mapping from the native endpoint to Qlik Replicate data types, see the chapter for the endpoint you are using. For a list of supported databases, see <i>Supported Platforms and Endpoints (page 1313)</i>.</p>
Add a new column	<p>Click Add Column to add a new column. When you add a column, the Name is blank and the Type is listed as <code>string(50)</code>.</p> <p>Type a name for the new column in the Name column. If needed (according to the column data), click in the Type column and select a data type from the list.</p>
Add an existing column	<p>From the Input pane, select one or more columns and click the right facing arrow button.</p> <p>To add all of the columns, click the right-facing double arrow.</p> <div data-bbox="507 1473 1390 1877" style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> • <i>By default all tables columns are included in the Output list. To include only some of the columns clear the By default include all columns check box at the top of the Transform tab. This removes all of the columns from the list. You can then add back any existing column as required.</i></p> <p>• <i>If a column is explicitly added at the table level but then dropped at the global level (using a global transformation rule), the column will still be created on the target, but without any data.</i></p> </div>

To	Do This
Delete a column	<p>From the Output list, select the row with the column you want to delete and click the left-facing arrow button.</p> <p>To remove all columns, click the left-facing double arrow. Note that all the columns except for columns defined as a primary key/unique index are deleted.</p>
Add/Remove a Unique Index segment to/from a target column	<p>A key icon indicates which target columns segments of the Unique Index.</p> <p>To add a Unique Index segment, click in the Key column to the left of target column to which you want to add the segment. A key icon will appear.</p> <p>To remove a Unique Index segment, click the key icon to the left of the target column from which you want to remove the segment. The key icon will disappear.</p>
Recalculate the data for a column in the target endpoint	<p>Click in the Expression column in the row with the table column you want to change the data for. Enter an expression using SQLite syntax.</p> <p>See <i>Creating an expression for transformations (page 1091)</i> and <i>Using SQLite syntax with transformations (page 1091)</i> for information on creating expressions.</p> <p>Once you add a calculated expression, you can test the expression. See <i>Using the Expression Builder (page 1137)</i>.</p>
<p>Change the data type for a specific input column</p> <div data-bbox="204 1200 475 1563" style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p> Supported with the IBM DB2 for iSeries and IBM DB2 for z/OS source endpoints only.</p> </div>	<p>This is required if a source column is defined as character type but the data stored in that column is binary or vice versa.</p> <div data-bbox="507 1167 1390 1339" style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p> When the source column type is <i>STRING</i>, <i>WSTRING</i>, <i>CLOB</i>, or <i>NCLOB</i>, you must also select a Character Set, otherwise an error will be shown and the OK button will be disabled.</p> </div> <p>In the Input table, click the relevant cell in the Type column and then select either STRING or BYTES from the drop-down list as required.</p> <div data-bbox="507 1464 1390 1599" style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p> If you change a column's Type in the Input table, you also need to set the same Type for the corresponding column in the Output table.</p> </div> <p>Note that if you select <i>STRING</i>, you can also change the character set, as explained below.</p> <div data-bbox="507 1724 1390 1859" style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p> Modified cells will display a triangle in the top right corner. To see the original value, click the triangle.</p> </div>

To	Do This
<p>Change the Character Set for a specific input column</p> <div data-bbox="204 427 475 790" style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p> <i>Supported with the IBM DB2 for iSeries and IBM DB2 for z/OS source endpoints only.</i></p> </div>	<p>This is required if a source character column is wrongly encoded. For example, if a source character column is described as encoded in CCSID <i>x</i>, but the data stored in that column is actually encoded in CCSID <i>y</i>.</p> <p>You can also set a custom character set as described in <i>Setting a custom character set (page 1090)</i> below.</p> <p>In the Input table:</p> <ol style="list-style-type: none"> 1. Click the relevant cell in the Type column and select STRING from the drop-down list. 2. Click the relevant cell in the Character Set column and then select the appropriate character set from the drop-down list. <div data-bbox="587 757 1385 972" style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p> • <i>Only character sets compatible with the selected Type will be available for selection.</i></p> <p> • <i>Modified cells will display a triangle in the top right corner. To see the original value, click the triangle.</i></p> </div>


For a description of the various list actions that you can perform, see *List actions (page 96)*.

Setting a custom character set

The following procedure is supported with the IBM DB2 for iSeries and IBM DB2 for z/OS source endpoints only.

Perform the steps below if the source table is defined with an incorrect CCSID and the correct definition is actually in a UCM file.

1. Create a mapping data file with the file extension **.ucm**.

 *If you edit an existing UCM file, you must also change the values of the `<code_set_name>` and `<icu:alias>` properties. If the file does not contain an `<icu:alias>` property, then you only need to change the value of the `<code_set_name>` property.*

2. Create a CNV file for the UCM file by running the following command:

```
<product_dir>\bin\makeconv.exe -v <file_name>.ucm
```

Example:

```
"c:\Program Files\Attunity\Replicate\bin\makeconv.exe" -v 1047_EX.ucm
```

This will create a CNV file with the same name as the UCM file (for example, *1047_EX.cnv*).

3. Create a new subfolder named **icudt<XX>l** under `<product_dir>\bin` where the **XX** is the same as the number in the **icudt<XX>.dll** file name.

For example, If the DLL file name is **icudt69.dll**, create a new subfolder named **icudt69l**.

- Copy the CNV file to the subfolder you just created (<product_dir>\bin\icudt69l in the example above).



When using the Replicate File Channel, the file should be copied to the same location on both Replicate servers.

- Restart the Qlik Replicate UI Server service.
- Select the custom character set from the **Character Set** drop-down list; it will appear as the CNV file name followed by the word "Custom" e.g. **1047_EX.cnv (Custom)**.

Using a column's before-image data in a transformation

You can use a column's before-image data in a transformation. This is useful if you need to store the before-image data on the target.

To do this, simply specify the source column name in **Output** table's **Expression** column, in the following format:

```
$BI__MyColumn
```

Where **\$BI__** is a mandatory prefix (that instructs Replicate to capture the before-image data) and **MyColumn** is the source column name.

Although you can store the before-image data in an existing target column, it is recommended to create a new target column (using the **Add Column** button) in which to store the before-image data.

Creating an expression for transformations

Use an expression to define the contents of a new or re-calculated column.

To create an expression:


- In the **Transform** tab, select the row with the column for which you want to create an expression.
or
Click **Add Column** to add a new column.
- Click the *fx* button in the **Expression** column.
The **Expression Builder** opens.
- Build an expression as described in *Using the Expression Builder (page 1137)*.

Using SQLite syntax with transformations

The following table lists the SQLite operators that are supported with transformations.

SQLITE syntax operators

Operator	Description
	Concatenate strings. FIRST_NAME LAST_NAME PHONE_NUMBER <Office Only> (adds the string Office Only to the telephone number).

Operator	Description
+	Adds two values together. DEPARTMENT_ID+100 (adds 100 to each ID number). Any column used in an expression with this operator must be a numeric data type.
-	Subtracts a value from another value. MANAGER_ID-100 (subtracts 100 from each ID number). Any column used in an expression with this operator must be a numeric data type.
%	Uses the remainder of a division expression as the value. %SALARY/7 (Divides the value of the Salary column by 7 and uses any remainder from the expression as the column value).
/	Divides one value into another. SALARY/.16 (Divides the value of the Salary column by .16. <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;">  <i>If the two values in the division expression are integers (two NUMERIC columns with no digits after the decimal) and the result is a fractional value, the result returned will be 0.</i> </div>
*	SALARY*.16 (Multiplies the value of the Salary column by .16. This could be used to calculate taxes that are subtracted from a salary).

For more information about SQLite syntax, see the SQLite documentation.

Using filters

Filters let you include or exclude records from a replication task based on the value(s) of the source table columns, thereby allowing you to replicate only the specific data that you need.

In this section:

- *Filter limitations (page 1092)*
- *Opening the Filter tab (page 1093)*
- *Creating a filter condition for a specified column (page 1094)*
- *Creating a record selection condition for one or more columns (page 1094)*
- *Adding or removing filter ranges (page 1096)*
- *Using SQLite syntax with filtering (page 1097)*

Filter limitations

When creating a filter, the following limitations apply:

- Filters are not supported for calculating columns of Right-to-Left languages.
- Filters can only be applied to immutable columns.

- **Filters on mutable columns:**
When a filter is created to exclude/include specific rows in a column, the specified rows will always be excluded/included, even if the rows that were originally excluded/included are later changed. For example, if you chose to exclude/include rows "1-10" in a column named "Age" and those rows were later changed to "11-20", the rows will continue to be excluded/included, even though the data is no longer the same.
Additionally, if a row outside the filter scope was changed (i.e. updated or updated and then deleted) so that it should now be excluded/included (as defined by the filter), it will not be replicated to the target. So, for example if you created a filter to exclude/include rows less than 5 and then changed row 6 to -6, it will not be replicated (even though it is included in the filter's criteria range).
- Filter cannot be applied to LOB columns.
- When specifying numeric data as a filtering condition, the data preceding the decimal point cannot exceed int64.

Opening the Filter tab

The Filter Table tab contains the following information:

- **Data Columns** list: This list contains a list of the columns for the table where you filtering data. You can use these to select the columns to use in the filtering operations.
This list has the following tabs:
 - **Source:** This tab lists the original source columns in the table.
 - **Header:** This tab lists the available header columns. You can create filters using these columns and include them in expressions. For information on these header columns, see *Headers (page 1143)*.
 - **Calculated:** This tab lists the columns added to the table. You add columns through transformations. For more information, see *Defining transformations for a single table/view (page 1084)*.
- **Filter Conditions** table: This table has the following columns:
 - **Name:** The name of the column where you are filtering the data.
 - **Type:** The data type for the column.
 - **Include/Exclude:** Indicate whether to include or exclude the filtered data for this column.
 - **Ranges:** Click the button on the right of the **Ranges** field to open the Range Builder. For information on creating a value or ranges with the Range Builder, see *Adding or removing filter ranges (page 1096)*.
For more information on typing in the filter ranges manually, see *Using SQLite syntax with filtering (page 1097)*.
- **Record Selection Condition:** Enter a complex condition that can include multiple columns. The condition must evaluate to TRUE to be accepted. You can create a condition using SQLite operators or by *Using the Expression Builder (page 1137)*. For information on using the SQLite operators, see *Creating a record selection condition for one or more columns (page 1094)*.

The following figure is an example of the information in the **Filter** tab of the **Table Settings** window.

Table Settings: Filter

HR.DEPARTMENTS Table Settings

General
Transform
Filter

Data Columns

Source	Header	Calculated
DEPARTMENT_ID		
DEPARTMENT_NAME		
MANAGER_ID		
LOCATION_ID		

>
<
>

Filter Conditions

Name	Type	Include/Exclude	Ranges
DEPARTMENT_NAME	STRING(30)	Include	

Record Selection Condition

Expression Builder

Restore Table Defaults

OK Cancel

To open the Filter tab:

1. Select the table you want to filter and then open the *Table Settings* (page 1082) window.
2. Click the **Filter** tab on the left side of the window.

Creating a filter condition for a specified column

You can create a simple condition for a single column in the table you are working with. You can include any combination of ranges or specific values in the filter and determine whether to include or exclude the defined data.

To create a filter condition:

1. Select a column from the data columns list and then click the right-facing arrow next to the **Filter Conditions** table.
To remove the column, click on it in the **Filter Conditions** table and then click the left-facing arrow. Any data entered for this column in the **Include/Exclude** or Values columns is also deleted.
2. Click in the **Include/Exclude** column to select whether to include or exclude the data that meets this condition.
3. Click the Edit Ranges button in the **Ranges** column.
4. The **<Name> <Include/Exclude> Ranges** window opens. Continue from *Adding or removing filter ranges* (page 1096).

Creating a record selection condition for one or more columns

You can create a record selection condition manually and/or by using the Expression Editor.

When entering a string, you can use the following special characters:

- `%`: Matches any string of zero or more characters. For example, `Mc%` searches for every name that begins with **Mc** or `%bob%` includes every name that contains **bob**.
- `_`: Matches a single character (as a wildcard). For example: `'sm_th'` includes names that begin with **Sm** and end with **th**, such as **Smith** or **Smyth**. To search for an underscore character, use `[_]`.
- `[. .]`: Includes a range or set of characters. For example, `[CK]ars[eo]` includes names **Carsen**, **Karsen**, **Carson**, and **Karson** or `[M-Z]inger` includes all words that end in **inger** with the first letter between **M** and **Z**, such as **Ringer**, **Singer**, or **Zinger**.

For more information, see documentation on how to use Transact-SQL.

For information on what SQLite operators can be used to create Record Selection Condition filters, see *Using SQLite syntax with filtering (page 1097)*.

To create a record selection condition:

1. From the **Data Columns** list, select a source column, header column or calculated column and then click the arrow to the left of the **Record Selection Condition** pane.
2. Use SQLite operators, such as `<` or `=` to create the condition. Use any amount of strings or columns as you need to create a condition.

For example `$EMPLOYEE_ID < 100 AND $SALARY > 100,000`

In this case only rows that satisfy both of these conditions are replicated in the replication task.

The following example provides an example using SQL search pattern strings. Only rows that satisfy this condition are replicated.

```
$EMPLOYEE_NAME IS 'sm_th'
```

To create a record selection condition using the Expression Builder:

- Click **Open Expression Builder**. This button is located directly under the record selection condition box. Follow the directions for creating an expression in the section *Using the Expression Builder (page 1137)*.

Applying updates to specific columns only

You can define an expression that instructs Replicate only to apply UPDATES when a user-defined condition has been met, for example, only when specific columns have changed. This is useful in situations when there are many updates in the source that the user has deemed not relevant for the target, as "irrelevant" updates will be ignored.

Limitations

- Does not support columns that do not have Before-Image data (e.g. LOB columns)
- Does not support the following sources (i.e. sources that do not support Before-Image records):
 - ODBC with CDC
 - Teradata
- Subject to the existing expression builder and filter limitations
- Content-based filtering may result in loss of data or data corruption. For example, if the Primary Key value changes (an UPDATE operation), the expression may ignore the UPDATE if the columns that were specified in the expression did not change. The result in this case would be that a "phantom" row with

the old row contents will remain, even if a change was later applied to the columns specified in the expression.

How to

Assume that you have a source table named **table1** with columns **c1-c10** but you are only interested in updating columns **c7**, **c8** and **c9** on the target.

To do this, you would need to:

1. [Open the Table Settings](#) for **table1** and select the **Filter** tab.
2. Click the **Expression Builder** button at the bottom right of the tab.
The Expression Builder opens.
3. Optionally, select the **Headers** tab.



*Although selecting the **Headers** tab is optional, selecting it will enable you to add `$AR_H_OPERATION` to your expression (as required in Step 4 below) simply by double-clicking the column on the left of the tab.*

4. Enter the following expression in the **Build Expression** pane and then click **OK**:
`($AR_H_OPERATION != 'UPDATE') OR
 ((($AR_H_OPERATION = 'UPDATE') AND ((ifnull($BI__c7,0) != ifnull($c7,0)) OR (ifnull($BI__c8,0) != ifnull($c8,0)) OR (ifnull($BI__c9,0) != ifnull($c9,0))))`
 The above expression means that changes will be applied to **c7**, **c8** and **c9** only if one of the following is true:
 - The operation is not an UPDATE.
 - The value of **c7**, **c8** or **c9** has changed as the result of an UPDATE operation.



When used in an expression, Before-Image columns must be prefixed with `$BI__`. For operations other than UPDATE, the value of the specified columns will be NULL.

Adding or removing filter ranges

You can add one or more values to the Ranges column using the Range Builder. Values that match any of the ranges in the list are included in the replication.

You can also delete a filter range using the Range Builder.



Filter ranges that you enter manually are also displayed in the Filter Builder. You can use the Filter Builder to delete them.

To use the Range Builder:

1. In the **Filter** tab of the *Table Settings (page 1082)* window, select a column to filter. For more information, see *Using filters (page 1092)*.
2. Click the button to the right of the **Ranges** column.
The **Ranges Builder** opens.
3. Click **Add Range**. Select any of the following from the drop-down list displayed.
 - **Equal to:** Select **Equal to** to enter a single value. The following is displayed in the range list.
Equal to = [N]
Click the **[N]** and type a value in the field that is displayed.
When the value in the selected column equals the value you enter, the result is included or excluded in the replication task depending on the option selected in the **Include/Exclude** column.
 - **Between:** Click **Between** to enter a range of values. The following is displayed in the range list.
Between [N] - [N]
Click each **[N]** and type a value in the fields that are displayed.
When the column contains the values between the two values entered, the result is included or excluded in the replication task depending on the option selected in the **Include/Exclude** column.
 - **Less than or equal to:** Select **Less than or equal to** and enter a maximum value. The following is displayed in the range list.
Less than or Equal to =< [N]
Click the **[N]** and type a value in the field that is displayed.
When the value in the selected column is equal to or less than the value you enter, the result is included or excluded in the replication task depending on the option selected in the **Include/Exclude** column.
 - **Greater than or equal to:** Select **Greater than or equal to** and enter a minimum value. The following is displayed in the range list.
Greater than or Equal to => [N]
Click the **[N]** and type a value in the field that is displayed.
When the value in the selected column is equal to or more than the value you enter, the result is included or excluded in the replication task depending on the option selected in the **Include/Exclude** column.

To delete a filter range from the Range Builder:

1. In the **Filter** tab of the *Table Settings (page 1082)* window, select the column with the filter condition you want to delete.
2. Click the button to the right of the **Ranges** column. The Ranges Builder opens.
3. Click the **X** next to the range you want to delete. The deleted range is removed from the list.

Using SQLite syntax with filtering

Qlik Replicate supports the following SQLite operators when creating Record Selection Condition filters.



You must put the (\$) in front of each input as shown below.

SQLITE syntax operators

Operator	Description
<	Is less than. <code>\$SALARY<100000</code>
<=	Is less than or equal to <code>\$SALARY<=100000</code>
>	Is greater than <code>\$SALARY>100000</code>
>=	Is more than or equal to <code>\$SALARY>=100000</code>
=	Is equal to <code>\$SALARY=100000</code>
!= or <>	Is not equal to <code>\$SALARY!=100000</code>
IS	Is the same as <code>\$HIRE_DATE IS 2014-09-29</code> IS functions the same as = unless one or both of the operands are NULL. In this case, if both operands are NULL, then the IS operator evaluates to 1 (true). If one operand is NULL and the other is not, then the IS operator evaluates to 0 (false).
IS NOT	Is not the same as <code>\$HIRE_DATE IS NOT 2014-09-29</code> IS NOT functions the same as != unless one or both of the operands are NULL. In this case, if both operands are NULL, the IS NOT operator evaluates to 0 (false). If one operand is NULL and the other is not, then the IS NOT operator evaluates to 1 (true).
AND	Both operands are true. <code>\$MANAGER_ID AND EMPLOYEE ID >100</code>
OR	Either operand is true. <code>\$MANAGER_ID OR EMPLOYEE ID >100</code>

For more information on how to use the SQLite syntax, see the SQLite documentation.

Parallel Load

In Full Load replication mode, you can accelerate the replication of large tables by splitting the table into segments and loading the segments in parallel. Tables can be segmented by data ranges, by partitions, or by sub-partitions.

Supported endpoints

The task must be defined with a combination of the following source and target endpoints:

Supported source endpoints:

- Amazon RDS for Microsoft SQL Server
- IBM DB2 for LUW
- IBM DB2 for z/OS



Table segmentation by partitions or sub-partitions is not supported with the IBM DB2 for z/OS source endpoint.

- Microsoft SQL Server
- MySQL
- Oracle
- PostgreSQL



Table segmentation by partitions or sub-partitions is not supported with the PostgreSQL source endpoint.

- SAP Sybase ASE
- SAP Application
- SAP Application (DB)



Tables are by default client dependent with the SAP Application (DB) source endpoint. The MANDT column is automatically taken directly from the endpoint.

- SAP HANA
- Teradata

Supported target endpoints:

- Amazon EMR
- Amazon MSK
- Amazon Redshift
- Amazon S3
- Cloudera Data Platform (CDP) Private Cloud
- Confluent Cloud
- Databricks (Cloud Storage)
- File
- Google Cloud BigQuery
- Google Cloud SQL for MySQL

- Google Cloud SQL for PostgreSQL
- Google Cloud Storage
- Google Dataproc
- Hadoop (Hortonworks, Cloudera, and MapR)
- Hortonworks Data Platform (HDP)
- Kafka
- Microsoft Azure ADLS
- Microsoft Azure Database for MySQL
- Microsoft Azure Database for PostgreSQL
- Microsoft Azure Data Warehouse
- Microsoft Azure HDInsight
- Microsoft Azure SQL Database
- Microsoft SQL Server
- MySQL
- Oracle
- PostgreSQL
- Snowflake on Google
- Snowflake on AWS
- Snowflake on Azure
- Sybase ASE
- Vertica

Setting up Parallel Load

To define segment boundaries by data range:

1. In the **Parallel Load** tab's **Select Parallel Load Method** section, select **Use Data Ranges**.
2. In the **Select Details** section, click **Select Segment Columns**.
The **Columns** window opens
3. For all endpoints, the Unique Index column is automatically selected. Select which additional columns whose data you wish to use to delineate the ranges and then click **OK**.



- *Selecting indexed columns will significantly improve performance*
- *You can select up to ten columns (multi-selection is supported)*
- *Records with null values will not be replicated*
- *The following data types cannot be used to define segments by ranges: DOUBLE, FLOAT, and LOB (BLOB, CLOB, NCLOB)*

4. In the **Define Segment Boundaries** section:
 - a. Click **Add Segment** to add a segment.
The columns that you selected will appear as table headings.
 - b. Enter the upper data range for the segment in the selected columns.



Values in DATE columns must be entered in the format supported by the source. For example, for an Oracle source, the correct format would be:

- ALTER SESSION SET NLS_DATE_FORMAT:
'YYYY-MM-DD HH24:MI:SS' (specifying YYYY-MM-DD only is also valid)
- ALTER SESSION SET NLS_TIMESTAMP_FORMAT:
'YYYY-MM-DD HH24:MI:SS.FF9'
- ALTER SESSION SET NLS_TIMESTAMP_TZ_FORMAT:
'YYYY-MM-DD HH24:MI:SS.FF9 TZH:TZM'

- Add additional segments as required.
 - Click **Validate** to validate that the specified data corresponds to the source column data type and that all of the defined segments contain values.
 - To delete a segment, select the desired segment and then click **Delete**.
- Click **OK** to save your settings.



When **Use Data Ranges** is selected, all of the table data will be replicated, even if data ranges are not defined for all of the columns.

Usage example

Let's assume that the following segments are defined in the **Define Segment Boundaries** table:

Example table data

Column_1	Column_2	Column_3
10	30	105
20	20	120
100	12	99

In this case, the following "WHERE" clauses will be created for each load segment:

- **Segment 1:** ((COL1 < 10) OR ((COL1 = 10) AND (COL2 < 30)) OR ((COL1 = 10) AND (COL2 = 30) AND (COL3 < 105)))
- **Segment 2:** NOT ((COL1 < 10) OR ((COL1 = 10) AND (COL2 < 30)) OR ((COL1 = 10) AND (COL2 = 30) AND (COL3 < 105))) AND ((COL1 < 20) OR ((COL1 = 20) AND (COL2 < 20)) OR ((COL1 = 20) AND (COL2 = 20) AND (COL3 < 120)))
- **Segment 3:** NOT ((COL1 < 20) OR ((COL1 = 20) AND (COL2 < 20)) OR ((COL1 = 30) AND (COL2 = 20) AND (COL3 < 120))) AND ((COL1 < 100) OR ((COL1 = 100) AND (COL2 < 12)) OR ((COL1 = 100) AND (COL2 = 12) AND (COL3 < 99)))
- **Segment 4:** NOT ((COL1 < 100) OR ((COL1 = 100) AND (COL2 < 12)) OR ((COL1 = 100) AND (COL2 = 12) AND (COL3 < 99)))

To define segment boundaries by all of the table partitions:

Only select this method if you are sure that the table is already partitioned.

1. In the **Parallel Load** tab's **Select Parallel Load Method** section, select **Use Partitions**.
2. In the **Select Partitions** section, select **Use all table partitions**. This will segment the table according to partitions that already exist in the source database.
3. Select one the following:
 - Use main partitions
 - Use sub partitions



This option will be disabled if the source database does not support sub-partitions.

4. Click **OK**.

To define segment boundaries by specific partitions:

Only select this method if you are sure that the table is already partitioned.

1. In the **Parallel Load** tab's **Select Parallel Load Method** section, select **Use Partitions**.
2. In the **Select Partitions** section, select **Specify partitions**. This will split the data according to the specified source partitions.



*When **Specify partitions** is selected, only the specified partitions will be replicated.*

3. Click **Add Partition**.
4. Specify the name of an existing partition or sub-partition.
5. If you specified the name of a sub-partition, select the check box in the **Sub-Partition** column.



The check box will be disabled if the source database does not support sub-partitions.

6. Add additional partitions/sub-partitions as required.
7. To delete a partition/sub-partition, select the partition/sub-partition and then click **Delete**.
8. Click **OK** to save your settings.

Adjusting the number of segments that can be loaded in parallel

You can increase or decrease the number of segments that will be loaded in parallel. For example, if you selected the **Use all table partitions** option and the source table has 20 partitions, increasing the default number of concurrent tasks (5) may improve performance.



Sub-tasks are allocated for each segment \ partition \ sub partition.

For example: If you select a table with 6 partitions and load the table using the **Use Partitions** method, 5 partitions will be loaded in parallel, corresponding with the default number of concurrent tasks (5). When one of the sub-tasks completes its run, it will be assigned to loading the sixth partition.

The currently set value is displayed at the bottom of the **Parallel Load** tab. You can modify this value in the **Maximum number of tables to load in parallel** field in the *Full Load Tuning* (page 1174) tab.

Handling LOB columns

You can override the [task's LOB settings](#) for individual tables.




This option is only available for tasks defined with any combination of the following source and target endpoints: Oracle source, Oracle target, PostgreSQL source, PostgreSQL target, Microsoft SQL Server source, Microsoft SQL Server target, MySQL source, and MySQL target.







- During CDC or during Full Load when the **Allow unlimited LOB size** option is enabled, LOB data types are supported only in tables with a primary key or unique index.
- When replicating from Microsoft SQL Server, inline LOBS will always be read directly from the logs (i.e. without lookup).

The following LOB handling options are available:

LOB handling options

Option	Description
Replicate LOB columns	<p>When this option is selected (the default), LOB columns will be replicated.</p> <p>Note that replicating LOBs may impact performance. This is especially true in the case of the large LOBs which require Replicate to perform a lookup from the source table in order to retrieve the source LOB value.</p>
Allow unlimited LOB size	<p>Select this option - also known as Full LOB mode - to ensure that all LOBs are replicated without being truncated. This option should be selected when all (or nearly all) of the LOBs you wish to replicate are large (i.e. exceed 1 GB).</p> <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;"> <p> Note If the task's Change Processing Mode is set to "Batch optimized apply" (the default), Replicate will switch to "Transactional apply" mode to apply tables with LOBs.</p> </div>

Option	Description
Optimize handling when LOB size is less than (KB)	<p>Select this option when you need to replicate both small and large LOBs, and most of the LOBs are small.</p> <div data-bbox="399 389 1385 667" style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p> <i>This option is supported with the following endpoints only:</i></p> <ul style="list-style-type: none"> • Sources: Oracle, Microsoft SQL server, MySQL, PostgreSQL, IBM DB2 for LUW, and Sybase ASE. • Targets: Oracle, Microsoft SQL Server, MySQL, PostgreSQL, IBM DB2 for z/OS, and Sybase ASE. </div> <p>When this option is selected, during Full Load, the small LOBs will be replicated "inline" (which is more efficient), and the large LOBs will be replicated by performing a lookup from the source table.</p> <p>During Change Processing, however, both small and large LOBs will be replicated by performing a lookup from the source table.</p> <div data-bbox="399 936 1385 1205" style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p> <i>When this option is selected, Replicate will check all of the LOB sizes to determine which ones to transfer "inline". LOBs larger than the specified size will be replicated using Full LOB mode.</i></p> <p><i>Therefore, if you know that most of the LOBs are larger than the specified setting, it is better to use the Allow unlimited LOB size option instead.</i></p> </div>
Chunk size (KB)	<p>Optionally, change the size of the LOB chunks to use when replicating the data to the target. The default chunk size should suffice in most cases, but if you encounter performance issues, adjusting the size may improve performance.</p> <div data-bbox="399 1366 1385 1529" style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p> <i>With some databases, data type validation occurs when the data is inserted or updated. In such cases, replication of structured data types (e.g. XML, JSON, GEOGRAPHY, etc.) may fail if the data is bigger than the specified chunk size.</i></p> </div>

Option	Description
Limit LOB size to (KB)	<p>Select this option if you only need to replicate small LOBs or if the target endpoint does not support unlimited LOB size. The maximum permitted value for this field is 102400 KB (100 MB).</p> <p>When replicating small LOBs, this option is more efficient than the Allow unlimited LOB size option since the LOBs are replicated "inline" as opposed to via "lookup" from the source. During Change Processing, small LOBs are usually replicated via "lookup" from the source.</p> <p>As the value of the Limit LOB size to is in bytes, the size should be calculated according to the following formulas:</p> <ul style="list-style-type: none"> • BLOB – The length of the largest LOB. • NCLOB – The length of the longest TEXT <i>in characters</i> multiplied by two (as each character is handled as a double-byte). If the data includes 4-byte characters, multiply it by four. • CLOB – The length of the longest TEXT in characters (as each character is handled as a UTF8 character). If the data includes 4-byte characters, multiply it by two. <div style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> • Any LOBs larger than the specified size will be truncated.</p> <p>• During Change Processing from Oracle source, inline BLOBs are replicated inline.</p> <p>• Changes to this setting will only affect existing tables after they are reloaded.</p> </div>



*In some scenarios, tasks configured to replicate tables with multiple LOB columns may consume a large amount of memory. This is because Replicate allocates memory by multiplying the **Limit LOB size to** value by the [Commit rate during full load](#) value, the sum of which, it multiplies by the number of LOB columns being replicated. So, for example, if LOB size is limited to 5 MB and the default commit rate is used (10000 events), a task replicating 6 LOB columns will consume 300 GB of memory. Note that other factors such as the database type and version may also affect memory consumption.*

Should you encounter memory consumption issues and suspect that a combination of the above factors may be the cause, stop the task and lower the value in the [Commit rate during full load](#) field. Then resume the task. Repeat this process until acceptable performance/memory levels are reached.

These instructions apply to Change Processing and Full Load tasks.



Changes to a column's LOB size while a task is running will not be reflected in the Change Table, unless the target tables are created by Qlik Replicate. In such cases, the task must be configured to drop and create the Change Table (the default) and the target tables need to be reloaded (after the LOB size has changed).

For more information on the Change Table, see *Store Changes Settings* (page 1175). For information on reloading target tables, see *Working with tasks at runtime* (page 1203).

Message format

This tab is only available for tasks defined with a supported streaming endpoint.

When a task is defined with such an endpoint, you can specify a custom message format that will override the default Replicate message format. This may be useful if the consumer application needs to process the message in a particular format.

The custom message format can be defined at task level and/or at table level. When it is defined at both task *and* table level, the message format defined for the table will take precedence over the message format defined for the task.

To define a custom message at table level:

1. Select a table.
2. Open the **Table Settings** window as described in *Table Settings* (page 1082).
3. Select the **Message Format** tab and click the **Change to Table Policy** button.
4. Configure the message format as described in *Message Format* (page 1196).
5. To use the message format defined for the task, click the **Change to Task Policy** button.

For information on defining a custom message at task level, see *Message Format* (page 1196).

Full Load

This tab is available for tasks defined with the IBM DB2 for z/OS and IBM DB2 for iSeries source endpoints only.

Select the **Eliminate creation of duplicate records on full load** option if you need to prevent duplicate records from being replicated during Full Load. You can either set the option at task level or per table.

Note that selecting this option could impact performance as Replicate instructs the source database to return the table records by Primary Key order and then removes any duplicate records.

To prevent creation of duplicate records per table:

1. Select the desired table and then open the **Table Settings** window as described in *Table Settings* (page 1082).
2. Select the **Full Load** tab and click the **Change to Table Policy** button.
3. Select the **Prevent creation of duplicate records on the target** check box.

For information on preventing creation of duplicate records at task level, see *Full Load Settings* (page 1171).

13.2 Defining global rules

Global rules are a useful way of making changes across multiple tables and columns in the same task. You can define *transformation rules* that modify the source data or metadata before it reaches the target, and/or you can define *filter rules* that determine which records will be replicated to the target.



Global rules are not available in a Log Stream Staging setup.

For information on Log Stream Staging, see [Using the Log Stream \(page 1072\)](#).

- **Transformations** - One way transformations can be used is to change the names of all tables in a task. You can change the names using wildcards and patterns. For example, you may want to change the names of the tables from `account_%` to `ac_%`. This is helpful when replicating data from a Microsoft SQL Server endpoint to an Oracle endpoint where the Microsoft SQL Server endpoint has a limit of 128 characters for a table name and the Oracle endpoint has a limit of 31 characters. You may also need to change a specific data type in the source to a different data type in the target for many or all of the tables in the task. Global transformation will allow you to accomplish this without having to define a transformation for each individual table.



Table-specific transformations override global transformations. For example, you can define a global transformation that changes the data type for all tables from `DATE` to `DATETIME(6)` and then define another transformation for a specific table that changes the data type from `DATE` to `STRING(50)`.

For information on defining a transformation for a specific table, see [Defining Transformations for a Single Table/View](#).

For an explanation of how to create global transformation, see *Starting the Global Transformation Rules wizard (page 1107)*.

- **Filters** - Use filter rules to determine which records will be replicated to the target. Filter can be based on column data (e.g. only replicate records where Age is greater than 18) or record attributes (e.g. only replicate UPDATED records).

For an explanation of how to create global filters, see *Starting the Global Filter Rules wizard (page 1131)*.

Starting the Global Transformation Rules wizard

You define global transformations using the Global Transformation Rules wizard.

To start the Global Transformations wizard:

1. Open the task for which you want to create a global transformations or a global filter. You can click **Open** above the Tasks list or double-click the task.
2. If you are not in the Designer mode, click **Designer** at the top right of the screen. For more information on the Task View and the Designer mode, see *Designer mode (page 93)*.

- In Designer mode, click **Global Rules**.
The [Global Rules](#) window opens.
- Click the **New Rule** toolbar button and select **Transformation**.
The **New Transformation Rule** wizard opens.
- Enter the information to define a global transformation rule. The first step is selecting the *Transformation type* (page 1108).

Limitations for global transformations

The following limitations apply to global transformations:

- Transformations are not supported for columns with Right-to-Left languages.
- Transformations cannot be performed on columns that contain special characters (e.g. #, \, /, -) in their name.
- The only supported transformation for columns that are mapped to BLOB/CLOB data types (by Replicate) is to drop the column on the target.
- Expressions must be written using SQLite syntax only.
- Changing a global transformation value will not reload affected tables automatically. The target must be reloaded manually for the changes to take effect.
For information on reloading the target, see *Using the Run button options* (page 1204).

Transformation type

In the **Transformation type** step of the **New Transformation Rule** wizard, you define the type of transformation you want to be performed.



You can only create one rule for each transformation type on the same object (e.g. a column). If you create multiple rules for a single transformation type on the same object, only the last rule you create will be valid. For example, if you create the following rules (in order) to rename a schema:

Rename Schema: Add Prefix

Rename Schema: Add Suffix

OR

Rename Column: Add Prefix

Rename Column: Add Suffix

Only the second rule (adding a suffix) will be executed.

To select the transformation type:

- Enter a name for the rule.
The name cannot exceed 32 characters, contain non-Latin characters, or contain any of the following characters: \:.*? "<>|
- Select one of the following:

Table or Schema:

- **Rename schema:** Select this to change the schema name for multiple tables. For example, if you want all HR tables to be renamed PERS.
- **Rename table:** Select this to change the name of multiple tables. For example, if you want all tables named SALARY to be called WAGES.

Tablespace:

- **Change table tablespace:** Select this to change the table tablespace on the target. You can change the table tablespace regardless of what objects it contains or you can specify a condition for it to be renamed. For example, change all table tablespaces that contain the table Emp1oyees in the schema company.
By default (i.e. when this option is not selected), the tables will either be created in the source table tablespace on the target (when replicating from an Oracle source) or in the default database tablespace (when replicating from any other source).



This option is only available for tasks with an Oracle target endpoint.

- **Change index tablespace:** Select this to change the index tablespace on the target. You can change the index tablespace regardless of what objects it contains or you can specify a condition for it to be renamed. For example, change all table tablespaces that contain the table Emp1oyees in the schema Company.
By default (i.e. when this option is not selected), the indexes will either be created in the source table tablespace on the target (when replicating from an Oracle source) or in the default database tablespace (when replicating from any other source).



This option is only available for tasks with an Oracle target endpoint.

Column:

- **Rename column:** Select this to change the name of multiple columns. For example, if you want to change all columns with word MINIMUM to MIN.
- **Add column:** Select this to add a column with a similar name to multiple tables.
- **Drop column:** Select this to drop a column with a similar name from multiple tables.
- **Convert data type:** Select this if you want to change a specific data type to a different one across multiple tables. For example, if you want to change all Integer data types to a string.
- **Replace column data:** Select this to replace column data across multiple target tables.



In homogeneous replication tasks (such as Oracle to Oracle), modifying a single table column (by changing the column data type or length for example), will break the homogeneity for the entire table.

Change Table:

Change Table transformations are only available when the [Store Changes](#) replication option is enabled. For more information on Change Tables, see *Using Change Tables (page 1283)*.

- **Rename Change Table:** Select this to rename the Replicate Change Table for all tables or for any table that matches the specified schema name and/or table name.
- **Rename Change Table schema:** Select this to change the schema under which the Replicate Change Table will be created, for all tables or for any table that matches the specified schema name and/or table name.

3. Click **Next** to proceed to the *Transformation scope* (page 1110) step.

Transformation scope


In the **Transformation scope** screen, you define which tables will be affected by the transformation. For example, you can apply the rule to all tables that contain the word SALARY as part of their name.





The options displayed in this screen depend on the selected [Transformation Type](#).

The following table describes all available options. The second column lists the Transformation Type where the option to be available.

Transformation conditions

Option	Transformation Type	Description
Schema name is like %	All	<p>Leave the % sign to include all schemas in your global transformation.</p> <p>Click the % sign to add a filter. In this case you can enter any name combination to include only that schema in your global transformation rule.</p> <p>For example, enter HR to include only tables that have the schema HR.</p> <p>You can use the % sign as a wildcard. For example, H% includes all tables with a schema that begins with the letter H, such as HR, HELLO, or HQ.</p> <p>The % wildcard can be used in any position. For example, if you use it at the beginning, %H, then all table names that end in H are included in the transformation rule. The % can also be used in a middle position.</p> <div data-bbox="767 976 1390 1339" style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> <i>If you are using an Oracle target, you must enter a schema that exists on the target endpoint. Qlik Replicate does not create new schemas on an Oracle endpoint. If you want to use a new schema for the target, create the schema on the Oracle endpoint before running the task. For more information, see Using Oracle as a target (page 966).</i></p> </div>

Option	Transformation Type	Description
Table tablespace is like %	Change table tablespace	<div data-bbox="767 293 1390 427" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;">  <i>This option is only available if the task is defined with an Oracle target endpoint.</i> </div> <p>Leave the % sign to include all table tablespace names in your global transformation.</p> <p>Click the % sign to add a filter. In this case, you can enter any name combination to include only the specified table tablespace in your global transformation rule.</p> <p>For example, enter SYSTEM to include only table tablespaces called SYSTEM.</p> <p>You can also use the % sign as a wildcard anywhere in the string. For example, H% includes all table tablespaces that begin with the letter "H" whereas %H includes all table tablespaces that end with the letter "H".</p>
Index tablespace is like %	Change index tablespace	<div data-bbox="767 965 1390 1099" style="border: 1px solid #ccc; padding: 5px; margin-bottom: 10px;">  <i>This option is only available if the task is defined with an Oracle target endpoint.</i> </div> <p>Leave the % sign to include all index tablespace names in your global transformation.</p> <p>Click the % sign to add a filter. In this case, you can enter any name combination to include only the specified index tablespace in your global transformation rule.</p> <p>For example, enter SYSTEM to include only index tablespaces called SYSTEM.</p> <p>You can also use the % sign as a wildcard anywhere in the string. For example, H% includes all index tablespaces that begin with the letter "H" whereas %H includes all index tablespaces that end with the letter "H".</p>

Option	Transformation Type	Description
Table name is like %	All	<p>Leave the % sign to include all table names in your global transformation rule.</p> <p>Click the % sign to add a filter. In this case you can enter any name combination to include only tables with that specific name in your global transformation rule.</p> <p>You can use the % sign as a wildcard. For example, J% includes all tables with a name that begins with the letter J, such as JOBS, JOBS_HISTORY, or JACKSONVILLE.</p> <p>The % wildcard can be used in any position. For example, if you use it at the beginning, %H, then all table names that end in H are included in the transformation rule. The % can also be used in a middle position.</p>
Column name is like %	Rename column Drop column Convert data type Replace column value	<p>Leave the % sign to include all column names in your global transformation rule.</p> <p>Click the % sign to add a filter. In this case you can enter any name combination to include only columns with that specific name in your global transformation rule.</p> <p>You can use the % sign as a wildcard. For example, N% includes all columns with a name that begins with the letter N, such as NAME, NAME_FIRST, or NAME_LAST.</p> <p>The % wildcard can be used in any position. For example, if you use it at the beginning, %IES, then all column names that end in with the string "IES" are included in the transformation rule. The % can also be used in a middle position.</p>
Data type is	Convert data type Replace column value	<p>Select a new data type from the drop-down list. Make sure that the data type you select is compatible with the data in that column.</p> <p>For a description of Qlik Replicate data types, see <i>Replicate data types (page 81)</i>.</p> <p>For information about data type mapping from the native endpoint to Qlik Replicate data types, see the chapter for the endpoint you are using. For a list of endpoints supported by Qlik Replicate, see <i>Supported Platforms and Endpoints (page 1313)</i>.</p>
Scope expression	All	Click Advanced options to define a scope expression using the Expression Builder.

After you complete defining the transformation rule definitions, click **Next** to go to the *Transformation action* (page 1114) step.



If the global transformation type you are defining is **Drop Column**, you do not need to create a [Transformation Rule](#). In this case, click **Finish** to add the rule to the [Global Rules](#) list.

Transformation action

In the **Transformation action** screen, you define what happens to the objects affected by the transformation rule. For example, you can define a new name for the affected objects or add a prefix to the table names. Only objects that fall within the *Transformation scope* (page 1110) will be affected.

The following transformation options are available:

- *Rename Schema* (page 1114)
- (page 1118)
- (page 1118)
- *Rename Table* (page 1118)
- *Rename Column* (page 1121)
- *Add column* (page 1124)
- *Drop Column* (page 1124)
- *Convert data type* (page 1124)
- *Rename Change Table schema* (page 1129)
- *Rename Change Table* (page 1126)

When done, click **Next**.

Limitations for transformation rules

The following limitations apply to transformation rules:

- Transformations are not supported for columns with Right-to-Left languages.
- Transformations cannot be performed on columns that contain special characters (e.g. #, \, /, -) in their name.
- The only supported transformation for columns that are mapped to BLOB/CLOB data types (by Replicate) is to drop the column on the target.
- Expressions must be written using SQLite syntax only.
- Changing a global transformation value will not reload affected tables automatically. The target must be reloaded manually for the changes to take effect.
For information on reloading the target, see *Using the Run button options* (page 1204).



The options displayed in this screen depend on the [Transformation Type](#) selected.

Rename Schema

If your transformation type is **Rename Schema**, you can do the following:

- *Rename schema to (string) (page 1115)*
- *Add a prefix or suffix (page 1115)*
- *Remove a prefix or suffix (page 1115)*
- *Replace a prefix or suffix with different characters (page 1116)*
- *Convert schema name to uppercase (page 1117)*
- *Convert schema name to lowercase (page 1117)*
- *Rename schema (Expression) (page 1117)*

Rename schema to (string)

Use the **Rename schema to: [string]** option to change the name of all table schemas that you defined in the *Transformation scope (page 1110)* step to a different name. For example, if you have a schema called `human_Resources` and want to change all instances of this name to `HR` then enter the string `HR`. You can enter any string in this field.

Add a prefix or suffix

Use the **Add a prefix or suffix** option to add additional characters to the beginning or end of the schema name for all schemas that fit the definition you created in the *Transformation scope (page 1110)* step. For example, if the schema name is `HR`, you can add a suffix, such as `TAR` or `_TAR` to the schema name for all tables with that schema name. In this case, the resulting schema name will be `HRTAR` or `HR_TAR`.



If you are using Oracle as your target endpoint, Qlik Replicate does not create a new schema. Therefore, the schema name that is the result of replacing a prefix or suffix with a different string of characters must exist in the Oracle target endpoint. If the resulting schema name does not exist, you must create the schema in the Oracle endpoint before carrying out this task.

For more information, see [Limitations \(page 968\)](#) for using Oracle as a Target.

To globally add a prefix or suffix

1. Select **Add <Prefix/Suffix> Insert Characters to matching schema names**.
2. Click the word **Prefix** or **Suffix** and select one of these two from the list.
3. Click **[string]** to activate the field.
4. Type the characters you want as the prefix or suffix. If you want to include an underscore or other legal character to separate the prefix/suffix from the original name, you must add it as part of the character string.
5. Click **Finish** to add the rule to the [Global Rules](#) list.

Remove a prefix or suffix

Use the **Remove a prefix or suffix** option to remove a string of characters from the beginning or end of a schema name for all schema that fit the definition you created in the *Transformation scope (page 1110)* step.

For example, you can use this option to remove the letters `_REV` from the schema name for all tables in the schema `HR_REV`. In this case the schema name in the target will be `HR`.



If you are using Oracle as your target endpoint, Qlik Replicate does not create a new schema. Therefore, the schema name that is the result of replacing a prefix or suffix with a different string of characters must exist in the Oracle target endpoint. If the resulting schema name does not exist, you must create the schema in the Oracle endpoint before carrying out this task.

For more information, see [Limitations \(page 968\)](#) for using Oracle as a Target.

To globally remove a prefix or suffix

1. Select **Remove <Prefix/Suffix> Insert Characters from matching schema names**.
2. Click the word **Prefix** or **Suffix** and select one of these two from the list.
3. Click **[string]** to activate the field.
4. Type the characters you want to remove. If you want to remove an underscore or other legal character from the original name, you must add it as part of the character string.
5. Click **Finish** to add the rule to the [Global Rules](#) list.

Replace a prefix or suffix with different characters

Use the **Replace a prefix or suffix** option to replace a string of characters with a different string of characters. You determine whether to replace the characters at the beginning or end of a schema name for all schema that fit the definition you created in the *Transformation scope (page 1110)* step.

For example, you can use this option to replace the letters `_ORIG` with `_REPL` in the schema name for all tables in the schema `HR_ORIG`. In this case the schema name in the target will be `HR_REPL`.



If you are using Oracle as your target endpoint, Qlik Replicate does not create a new schema. Therefore, the schema name that is the result of replacing a prefix or suffix with a different string of characters must exist in the Oracle target endpoint. If the resulting schema name does not exist, you must create the schema in the Oracle endpoint before carrying out this task.

For more information, see [Limitations \(page 968\)](#) for using Oracle as a Target.

To globally replace a prefix or suffix

1. Select **Replace <Prefix/Suffix> Insert Characters by Insert Characters for all matching schema names**.
2. Click the word **Prefix** or **Suffix** and select one of these two from the list.
3. Click the first **[string]** to activate the field.
4. Type the characters from the existing (source) schema that you want to replace. If you want to include an underscore or other legal character from the original name in the string that you want to replace, you must add it as part of the character string.
5. Click the second **[string]** to activate the field.
6. Type the characters you want to use in the target. These characters replace the original (source)

characters in the target.

7. Click **Finish** to add the rule to the [Global Rules](#) list.

Convert schema name to uppercase

Use the convert to uppercase option to convert all of the letters in a schema name to upper case. For example:

- Schema_cat, becomes SCHEMA_CAT
- schema_cat, becomes SCHEMA_CAT
- sChEMa_Cat, becomes SCHEMA_CAT

To globally change the schema name to all uppercase

1. Select **Convert schema name to uppercase**.
2. Click **Finish** to add the rule to the [Global Rules](#) list.

Convert schema name to lowercase

Use the convert to lowercase option to convert all of the letters in a schema name to lower case. For example:

- Schema_cat, becomes schema_cat
- SCHEMA_CAT, becomes schema_cat
- sChEMa_Cat, becomes schema_cat

To globally change the schema name to all uppercase

1. Select **Convert schema name to lowercase**.
2. Click **Finish** to add the rule to the [Global Rules](#) list.

Rename schema (Expression)

Use the **Rename schema to [expression]** option to change the name of all table schemas that you defined in the *Transformation scope* (page 1110) step to a different name. For example, if you have a schema called Human_Resources and want to change all instances of this name to HR.



If you are using Oracle as your target endpoint, Qlik Replicate does not create a new schema. Therefore, the schema name that is the result of replacing a prefix or suffix with a different string of characters must exist in the Oracle target endpoint. If the resulting schema name does not exist, you must create the schema in the Oracle endpoint before carrying out this task.

For more information, see Limitations (page 968) for using Oracle as a Target.

To globally change a schema name

1. Select **Rename schema to [expression]**
2. Click the button to the right of the **Rename schema** option to open the Expression Editor. For information on how to use the Expression Editor, see *Using the Expression Builder* (page 1137). Then go to step 4.

or

Click **[expression]** to activate the field and continue with step 3.

3. Type an SQLite expression or a string (in quotes) to rename the schema. For example:

- "New_Schema"
- 'PREF_'||\$SCHEMA_NAME_VAR||'_SUFF'

You can use the following variables in the SQLite expression:

- \$SCHEMA_NAME_VAR
- \$TABLE_NAME_VAR
- \$COLUMN_NAME_VAR
- \$COLUMN_DATATYPE_VAR

4. Click **Finish** to add the rule to the [Global Rules](#) list.

Change table tablespace

If your transformation type is **Change table tablespace**, you can change the table tablespace on an Oracle target. You can also set certain conditions that must exist in the source for the table tablespace to be changed. These include schema name, table name and table tablespace name.

For more information, see the following topics:

- *Transformation type (page 1108)*
- *Transformation action (page 1114)*

Change index tablespace

If your transformation type is **Change index tablespace**, you can change the index tablespace on an Oracle target. You can also set certain conditions that must exist in the source for the tablespace to be changed. These include schema name, table name and index tablespace name.

For more information, see the following topics:

- *Transformation type (page 1108)*
- *Transformation action (page 1114)*

Rename Table

If your transformation type is **Rename Table**, you can do the following:

- *Rename table to (string) (page 1119)*
- *Add a prefix or suffix (page 1119)*
- *Remove a prefix or suffix (page 1119)*
- *Replace a prefix or suffix with different characters (page 1119)*
- *Convert table name to uppercase (page 1120)*
- *Convert table name to lowercase (page 1120)*
- *Rename table (expression) (page 1120)*

Rename table to (string)

Use the **Rename table to: [string]** option to change the name of all tables that you defined in the *Transformation scope (page 1110)* step to a different name. For example, if you have a table called EMPLOYEE and want to change all instances of this name to EMP then enter the string EMP. You can enter any string in this field.

Add a prefix or suffix

Use the **Add a prefix or suffix** option to add additional characters to the beginning or end of the table name for all tables that fit the definition you created in the *Transformation scope (page 1110)* step. For example, if the table name is EMPLOYEES, you can add a suffix, such as TAR or _TAR to the table name for all tables with that table name. In this case, the resulting table name will be EMPLOYEESTAR or EMPLOYEES_TAR.

To globally add a prefix or suffix:

1. Select **Add <Prefix/Suffix> Insert Characters to matching table names.**
2. Click the word **Prefix** or **Suffix** and select one of these two from the list.
3. Click **[string]** to activate the field.
4. Type the characters you want as the prefix or suffix. If you want to include an underscore or other legal character to separate the prefix/suffix from the original name, you must add it as part of the character string.
5. Click **Finish** to add the rule to the [Global Rules](#) list.

Remove a prefix or suffix

Use the **Remove a prefix or suffix** option to remove a string of characters from the beginning or end of a table name for all tables that fit the definition you created in the *Transformation scope (page 1110)* step.

For example, you can use this option to remove the letters _REV from the table name for all tables with the name EMPLOYEES. In this case the table name in the target will be EMPLOYEES.

To globally remove a prefix or suffix:

1. Select **Remove <Prefix/Suffix> Insert Characters from matching table names.**
2. Click the word **Prefix** or **Suffix** and select one of these two from the list.
3. Click **[string]** to activate the field.
4. Type the characters you want to remove. If you want to remove an underscore or other legal character from the original name, you must add it as part of the character string.
5. Click **Finish** to add the rule to the [Global Rules](#) list.

Replace a prefix or suffix with different characters

Use the **Replace a prefix or suffix** option to replace a string of characters with a different string of characters. You determine whether to replace the characters at the beginning or end of a table name for all tables that fit the definition you created in the *Transformation scope (page 1110)* step.

For example, you can use this option to replace the letters _ORIG with _REPL in the table names for all tables called EMPLOYEE_ORIG. In this case the table name in the target will be EMPLOYEE_REPL.

To globally replace a prefix or suffix:

1. Select **Replace <Prefix/Suffix> Insert Characters by Insert Characters for all matching schema names**.
2. Click the word **Prefix** or **Suffix** and select one of these two from the list.
3. Click the first **[string]** to activate the field.
4. Type the characters from the existing (source) schema that you want to replace. If you want to include an underscore or other legal character from the original name in the string that you want to replace, you must add it as part of the character string.
5. Click the second **[string]** to activate the field.
6. Type the characters you want to use in the target. These characters replace the original (source) characters in the target.
7. Click **Finish** to add the rule to the [Global Rules](#) list.

Convert table name to uppercase

Use the convert to uppercase option to convert a table name to all upper case. For example:

- Table_cat, becomes TABLE_CAT
- table_cat, becomes TABLE_CAT
- taBLe_Cat, becomes TABLE_CAT

To globally change the table name to all uppercase:

1. Select **Convert table name to uppercase**.
2. Click **Finish** to add the rule to the [Global Rules](#) list.

Convert table name to lowercase

Use the convert to lowercase option to convert a table name to all lower case. For example:

- Table_cat, becomes table_cat
- TABLE_CAT, becomes table_cat
- taBLe_Cat, becomes table_cat

To globally change the table name to all lowercase:

1. Select **Convert table name to lowercase**.
2. Click **Finish** to add the rule to the [Global Rules](#) list.

Rename table (expression)

Use the **Rename table to [expression]** option to change the name of all tables that fit the definition you created in the *Transformation scope (page 1110)* step. For example, if you have a table called EMPLOYEE and want to change all instances of this name as defined in the previous step it to EMP.

To change the table name:

1. Select **Rename table to: [expression]**
2. Click the button to the right of the **Rename table** option to open the Expression Editor. For information on how to use the Expression Editor, see *Using the Expression Builder (page 1137)*. Then go to step 4.
or
Click **[expression]** to activate the field and continue with step 3.
3. Type an SQLite expression or a string (in quotes) to rename the table. For example:
 - "New_Table"
 - 'PREFIX_'||\$TABLE_NAME_VAR||'_SUFFIX'

You can use the following variables in the SQLite expression:

- \$SCHEMA_NAME_VAR
- \$TABLE_NAME_VAR
- \$COLUMN_NAME_VAR
- \$COLUMN_DATATYPE_VAR

Rename Column

If your transformation type is **Rename Column**, you can do the following:

- *Rename column to (string) (page 1121)*
- *Add a prefix or suffix (page 1121)*
- *Remove a prefix or suffix (page 1122)*
- *Replace a prefix or suffix with different characters (page 1122)*
- *Convert column name to uppercase (page 1122)*
- *Convert column name to lowercase (page 1123)*
- *Rename column (expression) (page 1123)*

Rename column to (string)

Use the **Rename column to: [string]** option to change the name of all columns that you defined in the *Transformation scope (page 1110)* step to a different name. For example, if you have a table called SALARY and want to change all instances of this name to EMP then enter the string SAL. You can enter any string in this field.

Add a prefix or suffix

Use the **Add a prefix or suffix** option to add additional characters to the beginning or end of the column name for all columns that fit the definition you created in the *Transformation scope (page 1110)* step. For example, if the column name is SALARY, you can add a suffix, such as TAR or _TAR to the table name for all tables with that table name. In this case, the resulting table name will be SALARYTAR or SALARY_TAR.

To globally add a prefix or suffix:

1. Select **Add <Prefix/Suffix> Insert Characters to matching column names.**
2. Click the word **Prefix** or **Suffix** and select one of these two from the list.
3. Click the **[string]** to activate the field.

4. Type the characters you want as the prefix or suffix. If you want to include an underscore or other legal character to separate the prefix/suffix from the original name, you must add it as part of the character string.
5. Click **Finish** to add the rule to the [Global Rules](#) list.

Remove a prefix or suffix

Use the **Remove a prefix or suffix** option to remove a string of characters from the beginning or end of a column name for all columns that fit the definition you created in the *Transformation scope* (page 1110) step.

For example, you can use this option to remove the letters `_REV` from the column name for all columns with the name `SALARY`. In this case the column name in the target will be `SALARY`.

To globally remove a prefix or suffix:

1. Select **Remove <Prefix/Suffix> Insert Characters from matching column names**.
2. Click the word **Prefix** or **Suffix** and select one of these two from the list.
3. Click **[string]** to activate the field.
4. Type the characters you want to remove. If you want to remove an underscore or other legal character from the original name, you must add it as part of the character string.
5. Click **Finish** to add the rule to the [Global Rules](#) list.

Replace a prefix or suffix with different characters

Use the **Replace a prefix or suffix** option to replace a string of characters with a different string of characters. You determine whether to replace the characters at the beginning or end of a column name for all columns that fit the definition you created in the *Transformation scope* (page 1110) step.

For example, you can use this option to replace the letters `_ORIG` with `_REPL` in the column names for all columns called `SALARY_ORIG`. In this case the column name in the target will be `SALARY_REPL`.

To globally replace a prefix or suffix:

1. Select **Replace <Prefix/Suffix> Insert Characters by Insert Characters for all matching schema names**.
2. Click the word **Prefix** or **Suffix** and select one of these two from the list.
3. Click the first **[string]** to activate the field.
4. Type the characters from the existing (source) column that you want to replace. If you want to include an underscore or other legal character from the original name in the string that you want to replace, you must add it as part of the character string.
5. Click the second **[string]** to activate the field.
6. Type the characters you want to use in the target. These characters replace the original (source) characters in the target.
7. Click **Finish** to add the rule to the [Global Rules](#) list.

Convert column name to uppercase

Use the convert to uppercase option to convert a column name to all upper case. For example:

- Column_cat, becomes COLUMN_CAT
- column_cat, becomes COLUMN_CAT
- coLUMnM_Cat, becomes COLUMN_CAT

To globally change the table name to all uppercase

1. Select **Convert column name to uppercase**.
2. Click **Finish** to add the rule to the [Global Rules](#) list.

Convert column name to lowercase

Use the convert to lowercase option to convert a column name to all lower case. For example:

- Column_cat, becomes column_cat
- column_cat, becomes column_cat
- coLUMnM_Cat, becomes column_cat

To globally change the column name to all lowercase:

1. Select **Convert column name to lowercase**.
2. Click **Finish** to add the rule to the [Global Rules](#) list.

Rename column (expression)

Use the **Rename column to [expression]** option to change the name of all tables that fit the definition you created in the *Transformation scope* (page 1110) step. For instance, if you have a column called SALARY and want to change it to SAL.

To change the column name:

1. Select **Rename column to: [expression]**
2. Click the button to the right of the **Rename column** option to open the Expression Editor. For information on how to use the Expression Editor, see *Using the Expression Builder* (page 1137). Then go to step 4.
or
Click **[expression]** to activate the field and continue with step 3.
3. Type an SQLite expression or a string (in quotes) to rename the column. For example:

- "New_Column"
- 'PREF_'||\$COLUMN_NAME_VAR||'_SUFF'

You can use the following variables in the SQLite expression:

- \$SCHEMA_NAME_VAR
- \$TABLE_NAME_VAR
- \$COLUMN_NAME_VAR
- \$COLUMN_DATATYPE_VAR

Add column

When you add a column to multiple tables, you must provide a name, define the data type for the column and define the data that the column contains. The column that you define here is added to all tables that fit the definition you created in step *Transformation scope* (page 1110).

The following describes the information you must enter in the transformation rule page for adding a column.

- **Column name:** Click the **[string]** to activate the field. Type the name for the column in the field. A column with this name is added to all tables that fit the definition you created in step *Transformation scope* (page 1110).
- **Add to Primary Key:** Select to add the column to the target tables' Primary Key.
- **Expression:** Click the button to the right of this field to open the Expression Builder or type an expression using SQLite operators to define the data in the column. For information on how to use the Expression Editor to create an expression, see *Using the Expression Builder* (page 1137). For more information on creating expressions, see *Creating an expression for transformations* (page 1091) and *Using SQLite syntax with transformations* (page 1091).
- **Set target data type to:** Click the drop-down for a list of data types and select a new data type from the drop-down list. Make sure that the data type you select is compatible with the data in that column. When **BYTES**, **STRING** or **WSTRING** is selected, specify a **Length** as well. When **NUMERIC** is selected, specify a **Precision and Scale** as well.
- **Subtype:** When **CLOB**, **NCLOB**, **STRING**, or **WSTRING** data types are selected, you can also set a data subtype. Select either **JSON** or **XML** from the **Subtype** drop-down list. Make sure that the data in the new column will be compatible with the selected subtype. The default is **Regular**, which means that the regular data type will be used without a subtype.

For a description of available data types, see *Replicate data types* (page 81). For information about data type mapping from the native endpoint to Qlik Replicate data types, see the section for the endpoint you use. For a list of supported databases, see *Supported Platforms and Endpoints* (page 1313).

Drop Column

This option does not require a transformation rule. For this option you complete the Global transformation rule after the *Transformation scope* (page 1110) step.

Convert data type

The data type that you define in this step is applied to all columns and tables that fit the definition you created in the *Transformation scope* (page 1110) step. Make sure that the data type you select is compatible with the data in the columns you defined.

- **Set target data type to** - If you change the target value type (e.g. string to numeric), you may also need to change the data type of the target columns as well. For the **BYTES**, **STRING**, and **WSTRING** data types, you can optionally specify the **Length** as well. If you leave the default value ("0"), Replicate will calculate the data type length based on the source column definition. You can also set the length using an expression. When you click the **fx** button to the right of the **Length** field, the Expression Builder opens showing the **Metadata** tab. For an explanation of the variables in this tab, see *Metadata (Global transformations only)* (page 1143).

Example:

The following expression multiplies the modified data type length by two.

```
$AR_M_MODIFIED_DATATYPE_LENGTH * 2
```

For the **NUMERIC** data type, you can optionally set the **Precision** and **Scale**. If you leave the default value ("0"), Replicate will calculate the precision and/or scale based on the source column value.

- **Subtype:** When **CLOB**, **NCLOB**, **STRING**, or **WSTRING** data types are selected, you can also set a data subtype. Select either **JSON** or **XML** from the **Subtype** drop-down list. Make sure that the new data in the column will be compatible with the selected subtype. The default is **Regular**, which means that the regular data type will be used without a subtype.

For a description of available data types, see *Replicate data types (page 81)*. For information about data type mapping from the native endpoint to Qlik Replicate data types, see the section for the endpoint you use. For a list of supported databases, see *Supported Platforms and Endpoints (page 1313)*.

Replace column value

Use the **Replace column value** transformation to replace the values in the source columns (set in the **Transformation scope**) with different values in the corresponding target columns.

The following options are available:

- **Replace target value with** - Create an expression for replacing the value in the source column values with a different value in the target columns. When you click the **fx** button to the right of the field, the Expression Builder opens showing the **Data** tab. For an explanation of the variables in this tab, see *Data (global transformations only) (page 1142)*.

Example:

The following expression appends the string "_new" to the original source column values.

```
$AR_M_SOURCE_COLUMN_DATA || '_new'
```

- **Set target data type to** - If you change the target value type (e.g. string to numeric), you may also need to change the data type of the target columns as well. For the **BYTES**, **STRING**, and **WSTRING** data types, you can optionally specify the **Length** as well. If you leave the default value ("0"), Replicate will calculate the data type length based on the source column definition. You can also set the length using an expression. When you click the **fx** button to the right of the **Length** field, the Expression Builder opens showing the **Metadata** tab. For an explanation of the variables in this tab, see *Metadata (Global transformations only) (page 1143)*.

Example:

The following expression multiplies the modified data type length by two.

```
$AR_M_MODIFIED_DATATYPE_LENGTH * 2
```

For the **NUMERIC** data type, you can optionally set the **Precision** and **Scale**. If you leave the default value ("0"), Replicate will calculate the precision and/or scale based on the source column value.

- **Subtype:** When **CLOB**, **NCLOB**, **STRING**, or **WSTRING** data types are selected, you can also set a data subtype. Select either **JSON** or **XML** from the **Subtype** drop-down list. Make sure that the new data in the column will be compatible with the selected subtype. The default is **Regular**, which means that the regular data type will be used without a subtype.

See also: *Using the Expression Builder (page 1137)*

Rename Change Table



This transformation is only available when the Store Changes replication option is turned on.

If your transformation type is **Rename Change Table**, you can do the following:

- *Rename Change Table to (string) (page 1126)*
- *Add a prefix or suffix (page 1126)*
- *Remove a prefix or suffix (page 1127)*
- *Replace a prefix or suffix with different characters (page 1127)*
- *Convert Change Table name to uppercase (page 1127)*
- *Convert Change Table name to lowercase (page 1128)*
- *Rename Change Table (expression) (page 1128)*



- *Globally renaming a Change Table will override the Change Table [suffix defined in the task settings](#).*
- *The Change Table name must be different from the source table names. Otherwise, a table error will occur.*

Rename Change Table to (string)

Use the **Rename Change Table to: [string]** option to change the name of all Change Tables that you defined in the *Transformation scope (page 1110)* step to a different name. For example, if you have a Change Table called **EMPLOYEE** and want to change all instances of this name to **EMP** then enter the string **EMP**. You can enter any string in this field.

Add a prefix or suffix

Use the **Add a prefix or suffix** option to add additional characters to the beginning or end of the Change Table name for all Change Tables that fit the definition you created in the *Transformation scope (page 1110)* step. For example, if the Change Table name is **EMPLOYEES**, you can add a suffix, such as **TAR** or **_TAR** to the Change Table name for all Change Tables with that name. In this case, the resulting Change Table name will be **EMPLOYEESTAR** or **EMPLOYEES_TAR**.

To globally add a prefix or suffix:

1. Select **Add <Prefix/Suffix> <String> to matching Change Table names**.
2. Click the word **Prefix** or **Suffix** and select one of these two from the list.

3. Click **[string]** to activate the field.
4. Type the characters you want as the prefix or suffix. If you want to include an underscore or other legal character to separate the prefix/suffix from the original name, you must add it as part of the character string.
5. Click **Finish** to add the rule to the [Global Rules](#) list.

Remove a prefix or suffix

Use the **Remove a prefix or suffix** option to remove a string of characters from the beginning or end of a Change Table name for all Change Tables that fit the definition you created in the *Transformation scope* (page 1110) step.

For example, you can use this option to remove the letters `_REV` from the Change Table name for all Change Tables with the name `EMPLOYEES`. In this case the Change Table name in the target will be `EMPLOYEES`.

To globally remove a prefix or suffix:

1. Select **Remove <Prefix/Suffix> <String> from matching Change Table names**.
2. Click the word **Prefix** or **Suffix** and select one of these two from the list.
3. Click **[string]** to activate the field.
4. Type the characters you want to remove. If you want to remove an underscore or other legal character from the original name, you must add it as part of the character string.
5. Click **Finish** to add the rule to the [Global Rules](#) list.

Replace a prefix or suffix with different characters

Use the **Replace a prefix or suffix** option to replace a string of characters with a different string of characters. You determine whether to replace the characters at the beginning or end of a Change Table name for all Change Tables that fit the definition you created in the *Transformation scope* (page 1110) step.

For example, you can use this option to replace the letters `_ORIG` with `_REPL` in the Change Table names for all Change Tables called `EMPLOYEE_ORIG`. In this case the Change Table name in the target will be `EMPLOYEE_REPL`.

To globally replace a prefix or suffix:

1. Select **Replace <Prefix/Suffix> <String> by <String> for all matching Change Table names**.
2. Click the word **Prefix** or **Suffix** and select one of these two from the list.
3. Click the first **[string]** to activate the field.
4. Type the characters from the existing (source) schema that you want to replace. If you want to include an underscore or other legal character from the original name in the string that you want to replace, you must add it as part of the character string.
5. Click the second **[string]** to activate the field.
6. Type the characters you want to use in the target. These characters replace the original (source) characters in the target.
7. Click **Finish** to add the rule to the [Global Rules](#) list.

Convert Change Table name to uppercase

Use the convert to uppercase option to convert a Change Table name to all upper case. For example:

- Table_cat, becomes TABLE_CAT
- Change Table_cat, becomes TABLE_CAT
- taBLe_Cat, becomes TABLE_CAT

To globally change the Change Table name to all uppercase:

1. Select **Convert Change Table name to uppercase**.
2. Click **Finish** to add the rule to the [Global Rules](#) list.

Convert Change Table name to lowercase

Use the convert to lowercase option to convert a Change Table name to all lower case. For example:

- Table_cat, becomes Change Table_cat
- TABLE_CAT, becomes Change Table_cat
- taBLe_Cat, becomes Change Table_cat

To globally change the Change Table name to all lowercase:

1. Select **Convert Change Table name to lowercase**.
2. Click **Finish** to add the rule to the [Global Rules](#) list.

Rename Change Table (expression)

Use the **Rename Change Table to [expression]** option to change the name of all Change Tables that fit the definition you created in the *Transformation scope (page 1110)* step. For example, if you have a Change Table called EMPLOYEE and want to change all instances of this name as defined in the previous step it to EMP.

To change the Change Table name:

1. Select **Rename Change Table to: [expression]**
2. Click the button to the right of the **Rename Change Table** option to open the Expression Editor. For information on how to use the Expression Editor, see *Using the Expression Builder (page 1137)*. Then go to step 4.
or
Click **[expression]** to activate the field and continue with step 3.
3. Type an SQLite expression or a string (in quotes) to rename the table. For example:

- "New_Change_Table_Name"
- 'PREF_'||\$AR_M_SOURCE_TABLE_NAME||'_SUFF'

You can use the following metadata in the SQLite expression:

- \$AR_M_SOURCE_COLUMN_DATATYPE
- \$AR_M_SOURCE_COLUMN_NAME
- \$AR_M_SOURCE_SCHEMA
- \$AR_M_SOURCE_TABLE_NAME

Rename Change Table schema



This transformation is only available when the Store Changes replication option is turned on.

If your transformation type is **Rename Change Table schema**, you can do the following:

- *Rename Change Table schema to (string) (page 1129)*
- *Add a prefix or suffix (page 1129)*
- *Remove a prefix or suffix (page 1129)*
- *Replace a prefix or suffix with different characters (page 1130)*
- *Convert Change Table schema name to uppercase (page 1130)*
- *Convert Change Table schema name to lowercase (page 1130)*
- *Rename Change Table schema (expression) (page 1131)*

Rename Change Table schema to (string)

Use the **Rename Change Table schema to: [string]** option to change the name of all Change Table schemas that you defined in the *Transformation scope (page 1110)* step to a different name. For example, if you have a Change Table schema called EMPLOYEE and want to change all instances of this name to EMP then enter the string EMP. You can enter any string in this field.

Add a prefix or suffix

Use the **Add a prefix or suffix** option to add additional characters to the beginning or end of the Change Table schema name for all tables that fit the definition you created in the *Transformation scope (page 1110)* step. For example, if the Change Table schema name is EMPLOYEES, you can add a suffix, such as TAR or _TAR to the Change Table schema name for all Change Table schemas with that Change Table schema name. In this case, the resulting Change Table schema name will be EMPLOYEESTAR or EMPLOYEES_TAR.

To globally add a prefix or suffix:

1. Select **Add <Prefix/Suffix> Insert Characters to matching Change Table schema names**.
2. Click the word **Prefix** or **Suffix** and select one of these two from the list.
3. Click **[string]** to activate the field.
4. Type the characters you want as the prefix or suffix. If you want to include an underscore or other legal character to separate the prefix/suffix from the original name, you must add it as part of the character string.
5. Click **Finish** to add the rule to the [Global Rules](#) list.

Remove a prefix or suffix

Use the **Remove a prefix or suffix** option to remove a string of characters from the beginning or end of a Change Table schema name for all tables that fit the definition you created in the *Transformation scope (page 1110)* step.

For example, you can use this option to remove the letters _REV from the Change Table schema name for all Change Table schemas with the name EMPLOYEES. In this case the Change Table schema name in the target will be EMPLOYEES.

To globally remove a prefix or suffix:

1. Select **Remove <Prefix/Suffix> <String> from matching Change Table schema names.**
2. Click the word **Prefix** or **Suffix** and select one of these two from the list.
3. Click **[string]** to activate the field.
4. Type the characters you want to remove. If you want to remove an underscore or other legal character from the original name, you must add it as part of the character string.
5. Click **Finish** to add the rule to the [Global Rules](#) list.

Replace a prefix or suffix with different characters

Use the **Replace a prefix or suffix** option to replace a string of characters with a different string of characters. You determine whether to replace the characters at the beginning or end of a Change Table schema name for all tables that fit the definition you created in the *Transformation scope* (page 1110) step.

For example, you can use this option to replace the letters `_ORIG` with `_REPL` in the Change Table schema names for all Change Table schemas called `EMPLOYEE_ORIG`. In this case the Change Table schema name in the target will be `EMPLOYEE_REPL`.

To globally replace a prefix or suffix:

1. Select **Replace <Prefix/Suffix> <String> by <String> for all matching schema names.**
2. Click the word **Prefix** or **Suffix** and select one of these two from the list.
3. Click the first **[string]** to activate the field.
4. Type the characters from the existing (source) Change Table schema name that you want to replace. If you want to include an underscore or other legal character from the original name in the string that you want to replace, you must add it as part of the character string.
5. Click the second **[string]** to activate the field.
6. Type the characters you want to use in the target. These characters replace the original (source) characters in the target.
7. Click **Finish** to add the rule to the [Global Rules](#) list.

Convert Change Table schema name to uppercase

Use the convert to uppercase option to convert a Change Table schema name to all upper case. For example:

- `Table_cat`, becomes `TABLE_CAT`
- `table_cat`, becomes `TABLE_CAT`
- `taBLe_Cat`, becomes `TABLE_CAT`

To globally change the Change Table schema name to all uppercase:

1. Select **Convert Change Table schema name to uppercase.**
2. Click **Finish** to add the rule to the [Global Rules](#) list.

Convert Change Table schema name to lowercase

Use the convert to lowercase option to convert a Change Table schema name to all lower case. For example:

- Table_cat, becomes table_cat
- TABLE_CAT, becomes table_cat
- taBLe_Cat, becomes table_cat

To globally change the Change Table schema name to all lowercase:

1. Select **Convert Change Table schema name to lowercase**.
2. Click **Finish** to add the rule to the [Global Rules](#) list.

Rename Change Table schema (expression)

Use the **Rename Change Table schema to [expression]** option to change the name of all tables that fall within the scope you defined in the *Transformation scope (page 1110)* step. For example, if you have a Change Table schema called EMPLOYEE and want to change all instances of the Change Table schema name as defined in the previous step to EMP.

To rename the Change Table schema:

1. Select **Rename Change Table schema to**.
2. Click the button to the right of the **Rename Change Table schema to** option to open the Expression Editor. For information on how to use the Expression Editor, see *Using the Expression Builder (page 1137)*.
or
Click **[expression]** to activate the field and continue with step 3.
3. Type an SQLite expression or a string (in quotes) to rename the table. For example:
 - "New_Change_Table_Schema"
 - 'PREF_'||\$AR_M_SOURCE_SCHEMA||'_SUFF'
 You can use the following metadata in the SQLite expression:
 - \$AR_M_SOURCE_COLUMN_DATATYPE
 - \$AR_M_SOURCE_COLUMN_NAME
 - \$AR_M_SOURCE_SCHEMA
 - \$AR_M_SOURCE_TABLE_NAME
4. When you're done, click **Next** to see a summary of your rule and replace the default name and description, or **Finish** to add the rule to the [Global Rules](#) list.

Starting the Global Filter Rules wizard

You define global filters using the Global Filter Rules wizard.

To start the Global Filters wizard:

1. Open the task for which you want to create a global transformations or a global filter.
You can click **Open** above the Tasks list or double-click the task.
2. If you are not in the Designer mode, click **Designer** at the top right of the screen.
For more information on the Task View and the Designer mode, see *Designer mode (page 93)*.
3. In Designer mode, click **Global Rules**.

The [Global Rules](#) window opens.

4. Click the **New Rule** toolbar button and select **Filter**.
The **New Filter Rule** wizard opens.
5. Enter the information to define a global transformation rule. The first step is selecting the *Filter type* (page 1132).

Limitations for global filters

- Filters cannot be applied to columns that contain special characters (e.g. #, \, /, -) in their name.
- When specifying numeric data as a filtering condition, the data preceding the decimal point cannot exceed int64.

Filter type

In the **Filter Type** screen of the **New Filter Rule** wizard, you define the type of filtering you want to be performed.

- **Filter by columns** - Only include records that match specific column data. For example, only include records where Birth Date is later than 02-Feb-2021.



Filtering by columns containing changeable values (e.g. Age) may result in inconsistent data on the target.

- **Filter by record attributes** - Only include records that match specific record attributes. For example, only include UPDATE or INSERTED records.

Filter scope

In the **Filter Scope** screen of the **New Filter Rule** wizard, you define which tables will be filtered. For example, you can limit the filter to all tables that contain the word SALARY as part of their name.




The options displayed in this screen depend on selected [filter type](#).

Scope options when filtering by columns

The following table describes the options that are available when filtering by columns.

Columns filtering scope


Option	Description
Schema name is like %	<p>Leave the % sign to apply the rule to all source schemas (the default).</p> <p>Alternatively, enter a custom string. The rule will only be applied to columns in schemas that match the specified string.</p> <p>For example, enter HR to include only columns that belong to schema HR.</p> <p>You can use the % symbol as a wildcard. For example, specifying H% will include all columns in tables that belong to schemas beginning with the letter H, such as HR, HELLO, or HQ.</p> <p>The % wildcard can be used in any position. For example, if you use it at the beginning (%H), then the rule will be applied to all columns in schemas that end with the letter "H". The % symbol can also be used in the middle of a string.</p> <div data-bbox="379 786 1390 1037" style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> <i>If you are using an Oracle target, you must enter a schema that exists on the target endpoint. Qlik Replicate does not create new schemas on an Oracle endpoint. If you want to use a new schema for the target, create the schema on the Oracle endpoint before running the task. For more information, see Using Oracle as a target (page 966).</i></p> </div>
Table name is like %	<p>Leave the % sign to apply the rule to all source tables (the default).</p> <p>Alternatively, enter a custom string. The rule will only be applied to columns in tables that match the specified string.</p> <p>You can use the % symbol as a wildcard. For example, specifying J% will include all columns in tables with names beginning with the letter J, such as JOBS, JOBS_HISTORY, or JACKSONVILLE.</p> <p>The % wildcard can be used in any position. For example, if you use it at the beginning (%H), then the rule will be applied to all columns in tables that end with the letter "H". The % symbol can also be used in the middle of a string.</p>
Column name is like %	<p>Leave the % sign to apply the rule to all source columns (the default).</p> <p>Alternatively, enter a custom string. The rule will only be applied to columns that match the specified string.</p> <p>You can use the % symbol as a wildcard. For example, specifying N% will include all columns with names beginning with the letter N, such as NAME, NAME_FIRST, or NAME_LAST.</p> <p>The % wildcard can be used in any position. For example, if you use it at the beginning (%n), then the rule will be applied to all columns that end with the letter "n". The % symbol can also be used in the middle of a string.</p>

Option	Description
Data type is	<p>Optionally, select a specific data type from the drop-down list. Make sure the data type you select is compatible with the data in that column.</p> <p>For a description of Qlik Replicate data types, see <i>Replicate data types (page 81)</i>.</p> <p>For information about data type mapping from the native endpoint to Qlik Replicate data types, see the chapter for the endpoint you are using. For a list of endpoints supported by Qlik Replicate, see <i>Supported Platforms and Endpoints (page 1313)</i>.</p>
Scope expression	Click Advanced options to define a scope expression using the expression builder.

Scope options when filtering by record attributes

The following table describes the options that are available when filtering by record attributes.

Record attributes filtering scope

Option	Description
Schema name is like %	<p>Leave the % sign to apply the rule to all source schemas (the default).</p> <p>Alternatively, enter a custom string. The rule will only be applied to columns in schemas that match the specified string.</p> <p>For example, enter HR to include only columns that belong to schema HR.</p> <p>You can use the % symbol as a wildcard. For example, specifying H% will include all columns in tables that belong to schemas beginning with the letter H, such as HR, HELLO, or HQ.</p> <p>The % wildcard can be used in any position. For example, if you use it at the beginning (%H), then the rule will be applied to all columns in schemas that end with the letter "H". The % symbol can also be used in the middle of a string.</p> <div style="border: 1px solid gray; padding: 10px; margin-top: 10px;"> <p> <i>If you are using an Oracle target, you must enter a schema that exists on the target endpoint. Qlik Replicate does not create new schemas on an Oracle endpoint. If you want to use a new schema for the target, create the schema on the Oracle endpoint before running the task. For more information, see <i>Using Oracle as a target (page 966)</i>.</i></p> </div>

Option	Description
Table name is like %	<p>Leave the % sign to apply the rule to all source tables (the default).</p> <p>Alternatively, enter a custom string. The rule will only be applied to columns in tables that match the specified string.</p> <p>You can use the % symbol as a wildcard. For example, specifying J% will include all columns in tables with names beginning with the letter J, such as JOBS, JOBS_HISTORY, or JACKSONVILLE.</p> <p>The % wildcard can be used in any position. For example, if you use it at the beginning (%H), then the rule will be applied to all columns in tables that end with the letter "H". The % symbol can also be used in the middle of a string.</p>
Scope expression	Click Advanced options to define a scope expression using the expression builder.

After defining the filter scope, click **Next** to proceed to the *Transformation action* (page 1114) screen.

Filter action

In the **Filter action** screen, you create an expression for selecting records based on their column data (when defining a **Filter by columns**) or based on their attributes (when defining a **Filter by record attributes**). Only records that fall within the *Filter scope* (page 1132) will be affected.

Filter action for columns

Use the Expression Builder to define a record selection expression based on column data. Click the **fx** button to open the Expression Builder.

The Expression Builder opens displaying the **Data** tab. Define an expression using the [available parameters](#).

Example (Assumes that "Age" is the column being filtered):

Create the following expression to select all records for citizens between the age of 50 and 80.

```
$AR_M_SOURCE_COLUMN_DATA > '60' AND $AR_M_SOURCE_COLUMN_DATA < '70'
```

Filter action for record attributes

Use the Expression Builder to define a record selection expression based on record attributes. Click the **fx** button to open the Expression Builder.

The Expression Builder opens displaying the **Headers** tab. Define an expression using the [available Header columns](#).



Header columns are not relevant for Full Load tasks. They are only relevant when the task's Apply Changes or Store Changes replication options are enabled.

Example:

Create the following expression to select UPDATED records only:

```
$AR_H_OPERATION='UPDATE'
```

See also: *Using the Expression Builder (page 1137)*

Managing global rules

The **Global Rules** window lists the names and descriptions of all rules that are defined for the Qlik Replicate instance you are working with. This is where you go to edit, delete, reposition, or activate/deactivate rules.

Editing global rules



You cannot change the name of a transformation rule

To edit a global rule:

1. In the **Global Rules** window, select the rule you want to edit.
2. Click the **Edit** button or double-click the rule.
The **Edit Existing Transformation Rule** wizard or **Edit Existing Filter Rule** wizard opens.
3. Make any changes you need in the wizard.
For information on how to configure a global transformation, see the following topics:
 - *Transformation type (page 1108)*
 - *Transformation scope (page 1110)*
 - *Transformation action (page 1114)*For information on how to configure a global filter, see the following topics:
 - *Filter type (page 1132)*
 - *Filter scope (page 1132)*
 - *Filter action (page 1135)*

Deleting global rules

To delete a global rule:

1. In the **Global Rules** window, select the rule you want to delete.
2. Click the **Delete** button.
3. When prompted for confirmation, click **OK**.
The rule is removed from the list and deleted from the system.

Positioning global rules

Rules are applied in the order they appear. To prevent conflicts therefore, ensure that rules are arranged in a logical order. You can use the up/down arrows to rearrange the rule order.

Activating or deactivating global rules

To activate or deactivate a rule, select or clear the check box in the **Activate** column as needed.

13.3 Using the Expression Builder

The Qlik Replicate Expression Builder provides an easy way to build an expression. It provides you with easy access to the required elements for your expression without having to type out any information manually. You access the Expression Builder through the dialog boxes where you define [Filters](#), *Defining transformations for a single table/view (page 1084)*, and Global Transformations when you do any of the following:

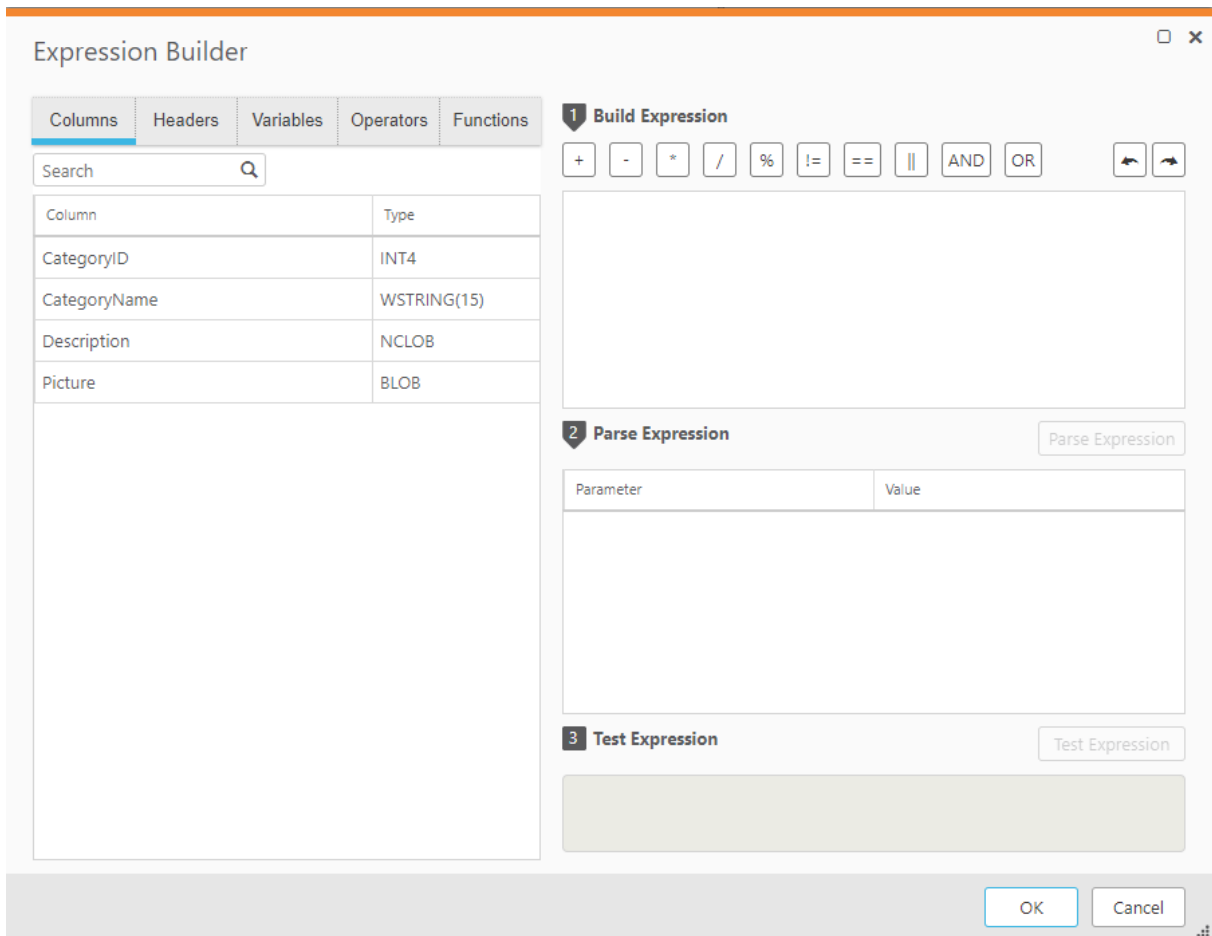
- *Rename Schema (page 1114)*
- *Rename Table (page 1118)*
- *Rename Column (page 1121)*

The following topics describe the Expression Builder:

- *Overview of the Expression Builder (page 1137)*
- *Build an expression (page 1139)*
- *Parse an expression (page 1140)*
- *Test an expression (page 1140)*
- *Using elements in the Expression Builder (page 1142)*

Overview of the Expression Builder

The following is an example of the Expression Builder with its four main parts shown. The Expression Builder you are working with may look different depending on whether you want to build an expression for a filter, a transformation, or a global transformation.



Expression Builder for filters, transformations, and global transformations

The following sections describe the tasks you can perform in each part of the Expression Builder:

- **Elements Pane (on the left):** This pane contains elements that you can add to an expression. Select elements and move them into the Expression Builder box to create the expression. For more information, see *Build an expression (page 1139)*.

The Elements Pane contains the following tabs:

- **Metadata** (available only when working with Global transformations)
- **Input** (available only when working with transformations or filters)
- **Header** (for Global transformations, this tab is available only when you select **Add Column**)
- **Variables**
- **Operators**
- **Functions**

For more information on these elements, see *Using elements in the Expression Builder (page 1142)*.

- **Build Expression Panel:** The Build Expression Panel is where you put together the expression you are building. You move elements, such as columns or operators into the box. You can also type all or part of an expression in this box.

For more information, see *Build an expression (page 1139)*.

- **Parse Expression Panel:** This panel displays the parameters for the expression. After you build the expression, click **Parse Expression** to list the expression parameters. You can then enter a value or argument for each of the parameters. For more information, see *Parse an expression (page 1140)*. The top part of the Expression panel contains the **Operator** toolbar. This toolbar contains the most common operators. Click the operator you want to use to add it to the expression. You can also add operators from the Element Pane, **Operators** tab.
- **Test Expression Panel:** This panel displays the results of a test that you can run after you provide values to each of the parameters in your expression. For more information, see *Test an expression (page 1140)*.

Build an expression

The first step in using the expression builder is to build an expression. The expression that you build is displayed in the top section of the right pane. You can open the Expression when:

- You define *Defining transformations for a single table/view (page 1084)* for a single table.
- You define [Filters](#) for a single table.
- You use the Global transformations dialog box to *Rename Schema (page 1114)*, *Rename Table (page 1118)*, *Rename Column (page 1121)*, or *Add column (page 1124)*.

Note: To add operators to your expression, you can use the **Operator** tab in the Element pane or the Operator buttons at the top of the Build Expression panel or any combination of these. See *Operators (page 1148)* and *Operator toolbar (page 1139)*.

For example, to create an expression that will combine the first name and last name, do the following:

1. In the **Input Columns** tab add the FIRST_NAME column to the **Build Expression** box.
2. Click the concatenate (||) operator from the **Operator** bar at the top of the Build Expression box.
3. In the Input Columns tab add the LAST_NAME column into the **Build Expression** box.

To build an expression:

1. In the Elements Pane, select any element you want to include in your expression. For information on the elements you can use in an expression, see *Functions (page 1151)*.
2. Add an element to the **Build Expression** panel by selecting it and then clicking the arrow to the right of the element.
3. Continue to add elements as needed.

Operator toolbar

The Operator toolbar is above the Build Expression box. It contains the most common operators so you can easily add them to an expression.

The following operators are available in the Operator toolbar: addition, subtraction, multiplication, division, percentage, not equal, equal, concatenate, AND, OR.



For information on these operators, see *Operators (page 1148)*.

To use the Operator toolbar:

1. Click the place in the Build Expression box where you want to add the operator.
2. Click the operator you want to add. It is added to the expression.

Parse an expression

You can parse an expression to determine its parameters and to determine whether the expression is valid.

To parse an expression:

1. In the Expression Builder window, create an expression as described in *Build an expression (page 1139)*.
2. Click **Parse Expression**.
If the expression is not valid, an error message is written in red at the bottom of the Expression Builder window.
If the expression is valid, the expression parameters are displayed in the **Parameter** column in the **Parse Expression** section. See the figure under *Test an expression (page 1140)*.
3. Type a valid value for each of the parameters in the **Value** column to *Test an expression (page 1140)*.
For example, type `John` for the `FIRST_NAME` and `Smith` for the `LAST_NAME` in the **Value** column. Once you type in values, you can *Test an expression (page 1140)*.

Test an expression

You can use the Qlik Replicate Test procedure to display the results of a test expression. The following figure is an example of a built expression that is evaluated and contains a test result.

1 Build Expression

+ - * / % != == || AND OR

`$First_Name || $Last_Name`

2 Evaluate Expression

Evaluate

Parameter	Value
<code>\$First_Name</code>	<input type="text" value="Mike"/>
<code>\$Last_Name</code>	<input type="text" value="Smith"/>

3 Test Expression

Test

MikeSmith

Test Expression dialog

To test an expression:

1. From the Expression Builder window, *Build an expression* (page 1139).
2. Click **Evaluate**. See *Parse an expression* (page 1140) for more information.

3. View the parameters that are displayed. If your expression is not valid, an error message is displayed. See *Parse an expression (page 1140)*.
4. Type values for each parameter then click **Test** to see the calculated expression. For example, type John for FIRST_NAME and Smith for LAST_NAME. The result displayed is JohnSmith. If you want a space between the words add it to the end of the FIRST_NAME value or the beginning of the LAST_NAME value.
Note: Testing calls to the source_lookup and target_lookup functions is not supported.

Using elements in the Expression Builder

You can use the following types of elements to build expressions for transformations, filters, and global transformations. Select the appropriate tab to select the elements.

- *Columns (transformations and filters only) (page 1142)*
- *Data (global transformations only) (page 1142)*
- *Metadata (Global transformations only) (page 1143)*
- *Variables (page 1147)*
- *Operators (page 1148)*
- *Functions (page 1151)*
- *Headers (page 1143)*
- *User-defined transformations (page 1163)*

Data (global transformations only)

In this tab, you can create an expression that leverages modified source column data and/or the original source column data. Modified column data is data that has been replaced or modified using a global transformation (e.g. **Replace column value**) whereas the original source column data is the data as it appears in the source database.

The following options are available:

- \$AR_M_MODIFIED_COLUMN_DATA - Use to specify column data that has been replaced or modified using a global transformation.
- \$AR_M_SOURCE_COLUMN_DATA - Use to specify the original column data.

Example:

For a source column called "Age", the following expression can be defined to exclude all records for citizens between the age of 50 and 80.

```
$AR_M_SOURCE_COLUMN_DATA > '50' AND $AR_M_SOURCE_COLUMN_DATA < '80'
```

Columns (transformations and filters only)

This tab lists the columns for the table you are working with. The table you are working with is the table you selected when you opened the **Table Settings** dialog box.



Expressions containing a condition for a `NUMBER` exceeding 19 digits are not valid. For example, the following expression is not valid:

```
case when $ID==12345678910111213149 then '1' else '0' end
```

Metadata (Global transformations only)

The **Metadata** tab contains the following variables that you can use in an expression. Variables with `MODIFIED` in their name can be used to specify metadata that has been replaced or modified using a global transformation (such as **Convert data type**) whereas variables names *without* `MODIFIED` can be used to specify the original metadata as it is defined in the source database.

- `AR_M_MODIFIED_SCHEMA` - The modified source schema name.
- `AR_M_MODIFIED_TABLE_NAME` - The modified source table name.
- `AR_M_MODIFIED_COLUMN_NAME` - The modified column name in the source table.
- `AR_M_MODIFIED_COLUMN_SUBTYPE_NAME` - The modified data type subtype of a column in the source table.
- `AR_M_MODIFIED_DATATYPE_NAME` - The modified data type of a column in the source table.
- `AR_M_MODIFIED_DATATYPE_LENGTH` - The modified data type length of a column in the source table.
- `AR_M_MODIFIED_DATATYPE_PRECISION` - The modified data type precision of a column in the source table.
- `AR_M_MODIFIED_DATATYPE_SCALE` - The modified data type scale of a column in the source table.
- `AR_M_SOURCE_SCHEMA` - The name of the source schema.
- `AR_M_SOURCE_TABLE_NAME` - The name of the source table.
- `AR_M_SOURCE_COLUMN_NAME` - The name of a column in the source table.
- `AR_M_SOURCE_DATATYPE_NAME` - The data type of a column in the source table.
- `AR_M_SOURCE_DATATYPE_LENGTH` - The data type length of a column in the source table.
- `AR_M_SOURCE_DATATYPE_PRECISION` - The data type precision of a column in the source table.
- `AR_M_SOURCE_DATATYPE_SCALE` - The data type scale of a column in the source table.

Example: Using metadata variables in the Rename Column transformation

To rename all columns named "product" to "source_schema.table_name", enter "product" in the **Column name is like** field (in the *Transformation scope* (page 1110) screen) and then enter the following expression in the **Rename column to** field (in the *Transformation action* (page 1114) screen):

```
$AR_M_SOURCE_SCHEMA || "." || $AR_M_SOURCE_TABLE_NAME
```

Headers

By default, headers for source tables are not replicated to the target. You can determine which, if any, headers to replicate when you define a transformation by creating an expression that includes the header.

You can create a filter using header values. Header filters are applied during change processing. See *Using filters* (page 1092) for additional information.



- The **Headers** tab in the Expression builder is available for filters and transformations. It is available for Global transformations only when you select **Add Columns**. See Transformation type (page 1108).
- Header columns are supported by all endpoints, except where explicitly stated to the contrary.

The available headers are described below.

AR_H_CHANGE_SEQ

Value in Change Processing

A monotonically increasing change sequencer that is common to all Change tables of a task. The Change sequence has the following format (with time being the UTC time on Replicate Server):

YYYYMMDDHHmmSShhxxxxxxxxxxxxxxxxxxxx

Where:

- YYYY is the four-digit year (such as 2012)
- MM is the two-digit month (range from 01-12)
- HH is the hour in the day (range from 00-23)
- mm is the minute in the hour (range from 00-59)
- SS is the second in the minute (range from 00-59)
- hh is the hundredth of the second (range from 00-99)
- xxxxxxxxxxxxxxxxxxx is a 19-digit, zero prefixed change number (global per task).

The time part usually refers to the commit time of the transaction that includes the change record. Qlik Replicate contains logic that maintains the monotonicity of the sequence number so modifying or adjusting the endpoint time may result in multiple changes to seem that they are within the same timestamp but with increasing change number.

The xxx...xxx is usually the internal change number from the data record except that for BEFORE-IMAGE records it is the same as the change number of the matching UPDATE record (for example, if the change number of BEFORE-IMAGE is 1000 and that of the UPDATE is 1001, then both have 1001). This allows a simple left-outer-join between the table and itself where on the left we scan until the point in time but filter out operation=before-image, and on the right we join on the same change_seq with the change_oper being 'B'.

Value in Full Load

Empty

Data type

VARCHAR(35)

AR_H_STREAM_POSITION

Value in Change Processing

The stream position value on the source - usually the SCN or LSN - which allows Replicate tasks to resume from the last processed event.

Value in Full Load

Empty string

Data type

STRING

AR_H_TIMESTAMP

Value in Change Processing

The timestamp of the Change operation, which is represented as the local time on Replicate Server. If your database and Replicate Server are located in the same timezone, the timestamp will be approximately representative of the actual Change operation.

Value in Full Load

Current timestamp

Data type

DATETIME

AR_H_TRANSACTION_ID

Value in Change Processing

The ID of the transaction to which the Change record belongs. The actual value is a hex-string of the 128-bit transaction ID. Depending on the source endpoint type, the ID may either be similar to the transaction ID in the source database or identical to it.



The transaction ID is not guaranteed to be unique and therefore should not be used to create reports or any other operation that relies upon its uniqueness.

Value in Full Load

Empty

Data type

VARCHAR (32)

AR_H_COMMIT_TIMESTAMP

Value in Change Processing

The source database commit timestamp, according to Replicate Server time. When the source database and Replicate Server are in the same timezone, the timestamp will be the approximate time of the actual commit on the source database.

Value in Full Load

Current timestamp

Data type

DATETIME

AR_H_DB_COMMIT_TIMESTAMP

Value in Change Processing

The source database commit timestamp, according to the database server time.



Relevant for the following source endpoints only: Oracle, Microsoft SQL Server, IBM DB2 for z/OS, Microsoft Azure SQL Managed Instance, and Amazon RDS for SQL Server.

Value in Full Load

NULL

Data type

DATETIME

AR_H_OPERATION

Value in Change Processing

Can be one of the following: INSERT, UPDATE, or DELETE.

Value in Full Load

INSERT

Data type

STRING

AR_H_USER

Value in Change Processing

The user that made the change to the source database.



Relevant for the following source endpoints only:

- *Microsoft SQL Server*
- *Microsoft Azure SQL Managed Instance*
- *Amazon RDS for SQL Server*
- *IBM Informix*
- *IBM DB2 for z/OS*
- *IBM DB2 for LUW*
- *Oracle*
- *File*
- [AIS-based sources](#)
- *SAP Sybase ASE*

Value in Full Load

Empty

Data type

STRING

AR_H_JOB_NAME

Value in Change Processing

The iSeries job that made the change to the source database.

*Relevant for the IBM DB2 for iSeries source endpoint only.***Value in Full Load**

Empty

Data type

STRING

AR_H_PROGRAM_NAME

Value in Change Processing

The iSeries program that made the change to the source database.

*Relevant for the IBM DB2 for iSeries source endpoint only.***Value in Full Load**

Empty

Data type

STRING

Variables

Your expression can contain any of the variables (which will be replaced during runtime) described in the table below.

Variables

Variable Name	Description	Data Type
AR_V_HOST_NAME	The host name of the machine on which Qlik Replicate Server is installed.	STRING (50)
AR_V_SOURCE_NAME	The logical name of the source endpoint defined in the endpoint settings.	STRING (50)
AR_V_TARGET_NAME	The logical name of the target endpoint defined in the endpoint settings.	STRING (50)

Variable Name	Description	Data Type
AR_V_TASK_NAME	The task name.	STRING (50)
AR_V_TASK_UUID	A unique string (Universal Unique Identifier) that identifies the task.	STRING (50)
AR_V_RELOAD_TIME	The time the source tables were reloaded.	DATETIME (6)
AR_V_START_TIME	The time the task started.	DATETIME (6)

Operators

The sections below describe the SQLite operators you can use to build an expression with the Expression builder. The Expression builder divides the operators into the following categories:

- *Strings (page 1148)*
- *Logical (page 1148)*
- *Mathematical (page 1151)*



With the exception of table-level transformations, all operator symbols must be preceded by a space and followed by a space. For example, the expression for concatenating a first and last name should be specified like this:

FIRST_NAME || LAST_NAME

And not like this:

FIRST_NAME||LAST_NAME

Strings

You can use the following string:

||

Name: Concatenate strings.

Examples:

FIRST_NAME || LAST_NAME

PHONE_NUMBER || <Office on1y> (adds the string office on1y to the telephone number).

Logical

The following table describes the logical SQLite operators used by the Qlik Replicate Expression Builder.

Logical operators

Operator	Description
!= or <>	Is not equal to <code>\$SALARY!=100000</code>
IS	Is the same as <code>\$HIRE_DATE IS 2014-09-29</code> IS functions the same as = unless one or both of the operands are NULL. In this case, if both operands are NULL, then the IS operator evaluates to 1 (true). If one operand is NULL and the other is not, then the IS operator evaluates to 0 (false).
IS NOT	Is not the same as <code>\$HIRE_DATE IS NOT 2014-09-29</code> IS NOT functions the same as != unless one or both of the operands are NULL. In this case, if both operands are NULL, the IS NOT operator evaluates to 0 (false). If one operand is NULL and the other is not, then the IS NOT operator evaluates to 1 (true).
IN	The IN operator takes a single scalar operand on the left and a vector operand on the right formed by an explicit list of zero or more scalars or by a single subquery. When the right operand of an IN operator is a subquery, the subquery must have a single result column. When the right operand is an empty set, the result of IN is false regardless of the left operand and even if the left operand is NULL. SQLite allows the parenthesized list of scalar values on the right-hand side of an IN operator to be an empty list but most other SQL endpoint engines and the SQL92 standard require the list to contain at least one element.
LIKE	The LIKE operator does a pattern matching comparison. The operand to the right of the LIKE operator contains the pattern and the left operand contains the string to match against the pattern. A percent symbol ("%") in the LIKE pattern matches any sequence of zero or more characters in the string. An underscore ("_") in the LIKE pattern matches any single character in the string. Any other character matches itself or its lower/upper case equivalent. (By default SQLite only understands upper/lower case for ASCII characters. The LIKE operator is case sensitive by default for unicode characters that are beyond the ASCII range. For example, the expression 'a' LIKE 'A' is TRUE but 'æ' LIKE 'Æ' is FALSE.) LIKE can be preceded by the NOT keyword.

Operator	Description
CASE	<p>Evaluates a list of conditions and returns one of multiple possible result expressions.</p> <p>Example 1:</p> <pre>WHEN \$NEWEST = 'Y' THEN '1' ELSE '0' END</pre> <p>Example 2:</p> <pre>case length(\$month) when 2 then \$year \$month when 1 then \$year 0 \$month end</pre>
GLOB	<p>The GLOB operator acts in the same way as the LIKE operator but uses the UNIX file globbing syntax for its wildcards. GLOB is case sensitive.</p> <p>GLOB can be preceded by the NOT keyword to invert the sense of the test. The infix GLOB operator is implemented by calling the function glob(Y,X) and can be modified by overriding that function.</p>
MATCH	<p>The MATCH operator is a special syntax for the match() application-defined function. The default match() function implementation raises an exception and is not really useful for anything. But extensions can override the match() function with more helpful logic.</p>
REGEXP	<p>The REGEXP operator is a special syntax for the regexp() user function. No regexp() user function is defined by default and so use of the REGEXP operator will normally result in an error message.</p>
AND	<p>Both operands are true.</p> <pre>\$MANAGER_ID AND EMPLOYEE ID >100</pre>
OR	<p>Either operand is true.</p> <pre>\$MANAGER_ID OR EMPLOYEE ID >100</pre>
<<	<p>Bitwise shift left.</p> <pre>x << n</pre> <p>A bitwise shift to the left of x by n bits.</p>
>>	<p>Bitwise shift right.</p> <pre>x >> n</pre> <p>A bitwise shift to the right of x by n bits.</p>
&	Unary and
	Unary or
<	<p>Is less than.</p> <pre>\$SALARY<100000</pre>

Operator	Description
<=	Is less than or equal to \$SALARY<=100000
>	Is greater than \$SALARY>100000
>=	Is more than or equal to \$SALARY>=100000
= or ==	Is equal to \$SALARY=100000

Mathematical

The following table describes the mathematical SQLite operators used by the Expression Builder.

Mathematical operators

Operator	Description
+	Adds two values together. DEPARTMENT_ID+100 (adds 100 to each ID number). Any column used in an expression with this operator must be a numeric data type.
-	Subtracts a value from another value. MANAGER_ID-100 (subtracts 100 from each ID number). Any column used in an expression with this operator must be a numeric data type.
%	Uses the remainder of a division expression as the value. %SALARY/7 (Divides the value of the Salary column by 7 and uses any remainder from the expression as the column value).
/	Divides one value into another. SALARY/.16 (Divides the value of the Salary column by .16. Note: If the two values in the division expression are integers (two NUMERIC columns with no digits after the decimal) and the result is a fractional value, the result returned will be 0.
*	SALARY*.16 (Multiplies the value of the Salary column by .16. This could be used to calculate taxes that are subtracted from a salary).

Functions

The sections below describe the SQLite functions you can use to build an expression with the Expression builder. The Expression builder divides the functions into the following categories:

- *Strings (page 1152)*
- *LOBs (page 1153)*
- *Numeric (page 1153)*
- *NULL check (page 1154)*
- *Date and Time (page 1154)*
- *Data Enrichment (page 1157)*
- *Operation (page 1160)*
- *Other Functions (page 1161)*
- *Hash (page 1162)*
- *User-defined transformations (page 1163)*

Strings

The following table describes the string functions used by the Expression Builder in Qlik Replicate.

String functions

Function	Description
lower(x)	The lower(x) function returns a copy of string x with all characters converted to lower case. The default built-in lower() function works for ASCII characters only.
ltrim(x,y)	The ltrim(x,y) function returns a string formed by removing all characters that appear in y from the left side of x. If there is no value for y, ltrim(x) removes spaces from the left side of x.
replace(x,y,z)	The replace(x,y,z) function returns a string formed by substituting string z for every occurrence of string y in string x.
rtrim(x,y)	The rtrim(x,y) function returns a string formed by removing all characters that appear in y from the right side of x. If there is no value for y, rtrim(x) removes spaces from the right side of x.
substr(x,y,z)	The substr(x,y,z) function returns a substring of input string x that begins with the y-th character and which is z characters long. If z is omitted then substr(x,y) returns all characters through the end of the string x beginning with the y-th. The left-most character of x is number 1. If y is negative then the first character of the substring is found by counting from the right rather than the left. If z is negative then the abs(z) characters preceding the y-th character are returned. If x is a string then characters indices refer to actual UTF-8 characters. If x is a BLOB then the indices refer to bytes.
trim(x,y)	The trim(x,y) function returns a string formed by removing all characters that appear in y from both sides of x. If there is no value for y, trim(x) removes spaces from both sides of x.

Function	Description
replaceChars (X,Y,Z)	<p>The <code>replaceChars (X, Y, Z)</code> function replaces any character in string <code>X</code> that also exists in string <code>Y</code> (characters to be replaced) with <code>Z</code> (replacement characters) in the same position. This is especially useful for removing non-valid characters from paths and file names.</p> <ul style="list-style-type: none"> • If string <code>Z</code> (replacement characters) does not include a character that has corresponding position in string <code>X</code>, it will be replaced with the first character in string <code>Z</code>. • If string <code>X</code> includes a character that does not exist in string <code>Z</code>, the original character will be left unchanged. <p>So, for example, specifying <code>replaceChars ("abcde", "abcd", "123")</code> would return <code>1231e</code>.</p>

LOBs

The following table describes the LOB functions used by the Expression Builder in Qlik Replicate.

LOB functions

Function	Description
hex(x)	The <code>hex()</code> function receives an argument as a BLOB and returns an upper-case hexadecimal string version of the BLOB content.
randblob (N)	The <code>randblob(N)</code> function returns an N-byte BLOB that contains pseudo-random bytes. If N is less than 1 then a 1-byte random BLOB is returned.
zeroblob(N)	The <code>zeroblob(N)</code> function returns a BLOB that consists of N bytes of 0x00.

Numeric

The following table describes the numeric functions used by the Expression Builder in Qlik Replicate.

Numeric functions

Function	Description
abs(x)	The <code>abs(x)</code> function returns the absolute value of the numeric argument <code>X</code> . <code>Abs(x)</code> returns NULL if <code>x</code> is NULL. <code>Abs(x)</code> returns 0.0 if <code>x</code> is a string or BLOB that cannot be converted to a numeric value.
random()	The <code>random()</code> function returns a pseudo-random integer between -9223372036854775808 and +9223372036854775807.
round(x,y)	The <code>round(x,y)</code> function returns a floating-point value <code>x</code> rounded to <code>y</code> digits to the right of the decimal point. If there is no value for <code>y</code> , it is assumed to be 0.

Function	Description
max(x,y,...)	The multi-argument max() function returns the argument with the maximum value, or returns NULL if any argument is NULL. The multi-argument max() function searches its arguments from left to right for an argument that defines a collating function and uses that collating function for all string comparisons. If none of the arguments to max() define a collating function, then the BINARY collating function is used. Note that max() is a simple function when it has two or more arguments but operates as an aggregate function if it has a single argument.
min(x,y,...)	The multi-argument min() function returns the argument with the minimum value. The multi-argument min() function searches its arguments from left to right for an argument that defines a collating function and uses that collating function for all string comparisons. If none of the arguments to min() define a collating function, then the BINARY collating function is used. Note that min() is a simple function when it has two or more arguments but operates as an aggregate function if it has a single argument

NULL check

The following table describes the NULL check functions used by the Expression Builder in Qlik Replicate.

NULL check functions

Function	Description
coalesce(x,y,...)	The coalesce() function returns a copy of its first non-NULL argument, it returns NULL if all arguments are NULL. Coalesce() have at least two arguments.
ifnull(x,y)	The ifnull() function returns a copy of its first non-NULL argument, it returns NULL if both arguments are NULL. Ifnull() must have exactly two arguments. The ifnull() function is the same as coalesce() with two arguments.
nullif(x,y)	The nullif(x,y) function returns a copy of its first argument if the arguments are different and returns NULL if the arguments are the same. The nullif(x,y) function searches its arguments from left to right for an argument that defines a collating function and uses that collating function for all string comparisons. If neither argument to nullif() defines a collating function then the BINARY is used.

Date and Time

The following table describes the Date and Time functions used by the Expression Builder in Qlik Replicate.

Date and Time functions

Function	Description
date(timestring, modifier, modifier,...)	Returns the date in the format YYYY-MM-DD.

Function	Description
<code>time(timestring, modifier, modifier,...)</code>	Returns the time in the format HH:MM:SS.
<code>datetime(timestring, modifier, modifier,...)</code>	Returns the date and time in the format YYYY-MM-DD HH:MM:SS.
<code>julianday(timestring, modifier, modifier,...)</code>	The <code>julianday()</code> function returns the number of days since noon in Greenwich on November 24, 4714 B.C.
<code>unixepoch(time-value, modifier, modifier,...)</code>	The <code>unixepoch()</code> function returns a unix timestamp - the number of seconds since 1970-01-01 00:00:00 UTC. The <code>unixepoch()</code> always returns an integer, even if the input time-value has millisecond precision.
<code>strftime(format, timestring, modifier, modifier...)</code>	<p>The <code>strftime()</code> routine returns the date formatted according to the format string specified as the first argument. It supports the following variables:</p> <ul style="list-style-type: none"> <code>%d</code>: day of month <code>%H</code>: hour 00-24 <code>%E</code>: ** fractional seconds SS.SSS <code>%j</code>: day of year 001-366 <code>%J</code>: ** Julian day number <code>%m</code>: month 01-12 <code>%M</code>: minute 00-59 <code>%s</code>: seconds since 1970-01-01 <code>%S</code>: seconds 00-59 <code>%w</code>: day of week 0-6 sunday==0 <code>%W</code>: week of year 00-53 <code>%Y</code>: year 0000-9999 <code>%%</code>: %

Examples

The expression builder provides you with a variety of options to build your own expression. You can use the regular operators with a date as well:

```
$HIRE_DATE < '2022-02-28'
```

Note that the right operand of the operator is contained between single quotes to prevent it being treated as a mathematical expression (2022 minus 2 minus 28).

Advanced examples using SQLite functions, modifiers and other operators

```
$HIRE_DATE < date('2022-02-28')
$DATE_1 < '2022-03-01' OR $DATE_1 > '2022-08-01'
$DATE_2 >= date('2022-03-01') AND $DATE_2 <= date('2022-08-01')
$HIRE_DATE < date('now','localtime','-1 year')
$HIRE_DATE <= date('now','utc')
$HIRE_DATE <= date('2022-02-28','+1 months')
$DATE_1 <= date($DATE_2,'+1 months')
```

List of valid modifiers in SQLite

The 'now' time string is the current date/datetime. You can specify an explicit date '2022-02-28' or you can use the date from a table column.

You can also apply multiple modifiers such as `datetime('now', '-3 hours', '+1 months')`, for example.

Modifier	Example	Result
	<code>datetime('now')</code>	2020-04-26 00:53:53
NNN days	<code>date('now', '+3 days')</code>	2020-04-29
NNN hours	<code>datetime('now', '-3 hours')</code>	2020-04-26 03:53:53
NNN minutes	<code>datetime('now', '+3 minutes')</code>	2020-04-26 00:56:53
NNN.NNNN seconds	<code>datetime('now', '-30 seconds')</code>	2020-04-26 00:54:23
NNN months	<code>date('now', '+3 months')</code>	2020-07-26
NNN years	<code>date('now', '-3 years')</code>	2017-04-26
start of month	<code>date('now', 'start of month')</code>	2020-04-01
start of year	<code>date('now', 'start of year')</code>	2020-01-01
start of day	<code>datetime('now', 'start of day')</code>	2020-04-26 00:00:00
weekday N	<code>date('now', 'weekday 6')</code>	2020-05-02
unixepoch	<code>datetime('1588965525', 'unixepoch')</code>	2020-05-08 19:18:45
localtime	<code>datetime('now', 'localtime')</code>	2020-04-26 10:53:53
utc	<code>datetime('now', 'utc')</code>	2020-04-25 08:53:53

Data Enrichment

Data Enrichment functions allow the selected source tables to be augmented with data from other records located in either the source or target endpoints. Practical applications of data enrichment functions include code lookup or master record lookup (e.g. social security number lookup to find a person's name).

You can enrich the target tables with supplemental data retrieved from the source or target endpoint by defining a transformation on the table. For more information about defining transformations on a single table, see *Defining transformations for a single table/view (page 1084)*.

Limitations

Amazon Redshift is not supported.

Data Enrichment functions

The table below describes the source and target lookup functions, which can be used both for table transformations and for global transformations. For a description of the parameters available for these functions, see Input Parameters below.

Data Enrichment functions

Function	Description
source_lookup (TTL, 'SCHM', 'TBL', 'EXP', 'COND', COND_PARAMS)	Use to retrieve additional data from the source endpoint.
target_lookup (TTL, 'SCHM', 'TBL', 'EXP', 'COND', COND_PARAMS)	Use to retrieve additional data from the target endpoint.

Input parameters

The possible input parameters for the lookup functions are described in the table below. For a usage example, see *Data Enrichment example (page 1159)*.

Data Enrichment input parameters

Function	Description
TTL	<p>TTL (Time to Live) is the amount of time the 'COND' return value will be cached. Caching the 'COND' return value improves performance by reducing the frequency that Qlik Replicate needs to access the source/target endpoint. As there is no default, you must specify a TTL value, which can be one of the following:</p> <p><SECONDS> - The time to cache the 'COND' return value in seconds. Specify a short caching time (e.g. 3) for data that is frequently updated or a long caching time for data that rarely changes.</p> <p>'NO_CACHING' - Specify 'NO_CACHING' if you do not want to cache the 'COND' return value. This is recommended for data that is constantly updated (e.g. share prices).</p> <p>'NO_EXPIRATION' - For data that is never updated (e.g. a street name), specify 'NO_EXPIRATION' to store the <i>Functions (page 1151)</i> return value permanently in the cache.</p>
'SCHM'	The schema name.
'TBL'	The table on which to perform the lookup.
'EXP'	<p>The expression to retrieve data from the lookup table.</p> <p>Note: The expression syntax must be native to the endpoint it accesses.</p> <p>The result should be a single column. Possible expressions include: <code>col1</code>, <code>col1+5</code>, <code>max (col1)</code>.</p> <p>Note: Full LOB columns are not supported. For information on including Limited-size LOB columns in the replication, see the description of the <i>Metadata (page 1166)</i> tab.</p> <p><i>Columns (transformations and filters only) (page 1142)</i>, <i>Headers (page 1143)</i>, and <i>Metadata (Global transformations only) (page 1143)</i> can also be used in the expression and are evaluated before the lookup statement is performed against the endpoint.</p>
'COND'	<p>The condition for the lookup statement.</p> <p>Note: The condition syntax must be native to the endpoint it accesses.</p> <p>The COND is a single field referencing all required fields.</p> <p>Example if the lookup table is located in Oracle:</p> <pre>'Fieldname1=:1 and Fieldname2=:2 and Fieldname3 =:3'</pre> <p>Example if the lookup table is located in Microsoft SQL Server:</p> <pre>'Fieldname1=? and Fieldname2=? and Fieldname3=?'</pre> <p><i>Columns (transformations and filters only) (page 1142)</i>, <i>Headers (page 1143)</i>, and <i>Metadata (Global transformations only) (page 1143)</i> can also be used in the expression and are evaluated before the lookup statement is performed against the endpoint.</p>

Function	Description
COND_PARAMS	<p>Any parameters required by the COND parameter.</p> <p>The COND_PARAMS (condition parameters) is not a single field, but a list of fields.</p> <p>Syntax:</p> <pre>\$FIELDNAME1 , \$FIELDNAME2 , \$FIELDNAME3</pre> <p>Full example:</p> <pre>source_lookup(10000 , 'HR' , 'DEPARTMENTS' , 'DEPARTMENT_NAME' , 'COMPANY_ID=? and DIVISION_ID=? and DEPT_ID=?' , \$COMP_ID , \$DIV_ID , \$DEPT_ID)</pre>

To improve efficiency, the source/target lookup tables should be indexed for the specified lookup fields.

Data Enrichment example

In the following example, Mike needs to add the DEPARTMENT_NAME column to the HR.JOB_HISTORY table. The DEPARTMENT_NAME column is located in the HR.DEPARTMENTS table in the source endpoint.

This is how the HR.JOB_HISTORY table appears before the column is added:

EMPLOYEE_ID	START_DATE	END_DATE	JOB_ID	DEPARTMENT_ID
102	2001-01-13 00:00:00	2006-07-24 00:00:00	IT_PROG	60
101	1997-09-21 00:00:00	2001-10-27 00:00:00	AC_ACCOUNT	110
101	2001-10-28 00:00:00	2005-03-15 00:00:00	AC_MGR	110
201	2004-02-17 00:00:00	2007-12-19 00:00:00	MK_REP	20
114	2006-03-24 00:00:00	2007-12-31 00:00:00	ST_CLERK	50
122	2007-01-01 00:00:00	2007-12-31 00:00:00	ST_CLERK	50
200	1995-09-17 00:00:00	2001-06-17 00:00:00	AD_ASST	90
176	0001-09-09 09:30:25	2010-09-09 09:30:00	SA_REP	60
176	2007-01-01 00:00:00	2010-09-09 09:30:00	SA_MAN	60
200	2002-07-01 00:00:00	2006-12-31 00:00:00	AC_ACCOUNT	90

This is how the HR.JOB_HISTORY table appears after the Full Load completes:

EMPLOYEE_ID	START_DATE	END_DATE	JOB_ID	DEPARTMENT_ID	DEPARTMENT_NAME
102	2001-01-13 00:00:00	2006-07-24 00:00:00	IT_PROG	60	Warehouse
101	1997-09-21 00:00:00	2001-10-27 00:00:00	AC_ACCOUNT	110	ProductManagement
101	2001-10-28 00:00:00	2005-03-15 00:00:00	AC_MGR	110	ProductManagement
201	2004-02-17 00:00:00	2007-12-19 00:00:00	MK_REP	20	HR
114	2006-03-24 00:00:00	2007-12-31 00:00:00	ST_CLERK	50	Bookkeeping
122	2007-01-01 00:00:00	2007-12-31 00:00:00	ST_CLERK	50	Bookkeeping
200	1995-09-17 00:00:00	2001-06-17 00:00:00	AD_ASST	90	Management
176	0001-09-09 09:30:25	2010-09-09 09:30:00	SA_REP	60	Warehouse
176	2007-01-01 00:00:00	2010-09-09 09:30:00	SA_MAN	60	Warehouse
200	2002-07-01 00:00:00	2006-12-31 00:00:00	AC_ACCOUNT	90	Management

To add the DEPARTMENT_NAME column, Mike needs to:

1. Create a new task and select the HR.JOB_HISTORY table for replication.
2. Apply a “New Column” transformation to the HR.JOB_HISTORY table. For more information on defining transformations, see *Defining transformations for a single table/view (page 1084)*.
3. Open the **Expression Builder** and choose **Data Enrichment** from the **Functions** tab. For more information on the Expression Builder, see *Using the Expression Builder (page 1137)*.
4. Select the `source_lookup` function and configure it as follows (using the native syntax of the source endpoint):

If the lookup table is located in Oracle:

```
source_lookup(10000, 'HR', 'DEPARTMENTS', 'DEPARTMENT_NAME',
'DEPARTMENT_ID=:1', $DEPARTMENT_ID)
```

If the lookup table is located in Microsoft SQL Server:

```
source_lookup
(10000, 'HR', 'DEPARTMENTS', '[DEPARTMENT_NAME]',
'[DEPARTMENT]=?', $DEPARTMENT_ID)
```

Where:


- 10000 is the TTL parameter.
- HR is the schema name.
- DEPARTMENTS is the table name.
- DEPARTMENT_NAME is the expression.
- DEPARTMENT_ID=:1 (or ? on Microsoft SQL Server) is the condition.
- \$DEPARTMENT_ID is the condition parameter.

5. Run the task.

Operation

The following table describes the Operation functions used by the Expression Builder in Qlik Replicate.

Operation functions

Function	Description
<code>operation_indicator (value_on_delete, value_on_update, value_on_insert)</code>	<p>When the <code>operation_indicator</code> function is invoked on its own or as part of an expression, records deleted from the source table will <i>not</i> be deleted from the corresponding target table. Instead, the corresponding target record will be flagged (with a user-provided value) to indicate that it was deleted from the source. The <code>operation_indicator</code> function also requires you to provide values to indicate records that were inserted or updated in the source endpoint.</p> <div data-bbox="421 577 1390 1039" style="border: 1px solid #ccc; padding: 10px; margin: 10px 0;"> <p> • <i>The <code>operation_indicator</code> function is not supported on tables that do not have a Primary Key.</i></p> <p>• <i>It is recommended to add a dedicated column for the flag values, for example, <code>OPERATION</code>. For an explanation of how to add a column, see Using the Transform tab (page 1086).</i></p> <p>• <i>This function is not supported when:</i></p> <ul style="list-style-type: none"> • <i>The Apply Conflicts error handling policy is set to No record found for applying an update: INSERT the missing target record.</i> • <i>The Apply changes using SQL MERGE task setting is enabled.</i> </div> <p>To specify the function values:</p> <p>Replace <code>value_on_delete</code>, <code>value_on_insert</code> and <code>value_on_update</code> with the values that you want to appear in the target endpoint.</p> <p>Values should be formatted according to the corresponding column type.</p> <p>Example when the column type is INT4:</p> <pre>operation_indicator('1', '0', '0')</pre> <p>Example when the column type is STRING:</p> <pre>operation_indicator('Deleted', 'Updated', 'Inserted')</pre>

Other Functions

The following table describes additional functions used by the Expression Builder in Qlik Replicate.

Other functions

Function	Description
length(x)	For a string value x, the length(x) function returns the number of characters (not bytes) in x before to the first NULL character. If x is NULL then length(x) is NULL. If x is numeric then length(X) returns the length of a string representation of X.
like(x,y,z)	The like() function is used to implement the "Y LIKE X [ESCAPE Z]" expression. The ESCAPE (z) clause is optional. If there is a z clause, then the like() function is invoked with three arguments. Otherwise, it is invoked with two arguments.
typeof(x)	The typeof(x) function returns a string that indicates the datatype of the expression x: null, integer, real, text, or BLOB.

Hash

The Hash function generates a hash value for an inputted column (using the SHA-256 algorithm) and then returns the hex value of the generated hash value.

To use the function in an expression, add the `hash_sha256(x)` function to the **Build Expression** pane and then replace the "x" with the desired source column name (from the **Input Columns** tab).

The function is especially useful for masking sensitive information. In the expression below, for example, the Hash function has been used to obfuscate employees' email addresses.

Expression Builder

Input Columns | Header Columns | Operators | Functions

Search

Column	Type
COMMISSION_PCT	NUMERIC(2,2)
DEPARTMENT_ID	INT2
EMAIL	STRING(25)
EMPLOYEE_ID	INT4
FIRST_NAME	STRING(20)
HIRE_DATE	DATETIME(6)
JOB_ID	STRING(10)
LAST_NAME	STRING(25)
MANAGER_ID	INT4
PHONE_NUMBER	STRING(20)
SALARY	NUMERIC(8,2)

1 Build Expression

+ - * / % != == || AND OR

hash_sha256(\$EMAIL)

2 Parse Expression Parse Expression

Parameter	Value
\$EMAIL	dan@company.com

3 Test Expression Test Expression

4D89E44A4ACBF7DE16F09EB5EE0D170B35127104127B168F6B949D296B1F147F

OK Cancel

User-defined transformations

Customers that requires functionality not provided by Replicate's built-in transformations can write their own transformations, and then access them from the [Replicate Expression Builder](#).

It is also recommended to use the Replicate Add-ons API which allows you to take advantage of Replicate's memory management and logging capabilities, while eliminating the need to create independent memory and logging routines.

For more information, see *Replicate add-ons API* (page 1334).

The procedure below is based on the sample files located in:

<INSTALL_DIR>\addons\samples\MyTransformation



The path is the same on Linux, but with slashes (/) instead of backslashes (\). Similarly, although the explanation below refers to a DLL file (Windows), on Linux this should be an SO file.

Changing the default addon name

You can change the default addon name (**MyTransformation**) simply by renaming the `<INSTALL_DIR>\addons\samples\MyTransformation\MyTransformation.dll` file as desired. Note that if you rename the DLL, you may also need to specify the new name when editing the **addons_def.json.sample** file described below (depending on which parameters you use).

Creating a user-defined transformation

To create a user-defined transformation:

1. Create a shared library that implements the following exported initialization function:

```
typedef int AR_ADDON_INIT_FUNC(AR_ADDON_CONTEXT *context);
```

All of the types and prototypes are defined in the **ar_addon.h** and

ar_addon_transformation.h files located under `<INSTALL_DIR>\addons\include`.

2. Make sure your shared library is in the following location:

```
<INSTALL_DIR>\addons\samples\addon_name
```



When working in a [High Availability](#) setup, the created binaries should be installed on all of the cluster instances.

3. Compile the transformation.

The DLL is automatically created in the following location:

```
<installation_dir>\addons\addon_name\addon_name.dll
```

4. Register the library in the **addons_def.json.sample** file located under `<INSTALL_DIR>\addons`.

```
{
  "addons": [{
    "name": "MyTransformation",
    "type": "STARTUP",
    //"lib_path": "C:\\Program Files\\Attunity
Replicate\\addons\\samples\\MyTransformation\\MyTransformation.dll",
    //"lib_path":
"/opt/attunity/replicate/addons/samples/MyTransformation/MyTransformation.so",
    "init_function": "my_transformation_init_func"
  }]
}
```

Where:

- `name` is the logical name of the DLL (can be any name). If you do not use the `lib_path` parameter to specify the DLL path (see below), then the DLL file must reside in `<INSTALL_DIR>\addons<addon_name>`. On Linux, the `<addon_name>` folder needs to be created manually and should either be the default addon name (**MyTransformation**) or its new name (if you changed it).
- `type` is an optional parameter that specifies when to load the DLL. The only value that is currently supported is `STARTUP`.

- `lib_path` is the full path of your DLL (e.g. `C:\Transformations\MyManipulator.dll`). This is required only if the DLL does not reside in `<INSTALL_DIR>\addons\<addon_name>` (as mentioned in the description of the name parameter above).
 - `init_function` is the function name, as it appears in the C file used to generate the DLL.
5. Save the file as **addons_def.json**.
 6. Register the new function in the addon initialization function (mentioned in Step 1) as in the following example:

```
USER_DEFINED_TRANSFORMATION_DEF *transdef = GET_AR_AO_TRANSFORMATION_DEF
();
transdef->displayName = "prefix_with(X, Y)";
transdef->functionName = "prefix_with";
transdef->description = "prefix_with adds the prefix <Y_> to a given
string X";
transdef->func = trans_prefix_with;
transdef->nArgs = 2;
AR_AO_REGISTRATION->register_user_defined_transformation(transdef);
```
 7. Restart the Qlik Replicate Server service.

The new "prefix_with" function will be available in the Expression Builder under **Functions->User Defined**.

13.4 Task Settings

Task-specific replication settings can be configured in the **<Task Name> - Settings** dialog box.



Some of the task settings are not available in a Log Stream Staging setup.

For information on the availability of task settings in a Log Stream Staging setup, see [Using the Log Stream \(page 1072\)](#).

To open the **<Task Name> - Settings** dialog box:

1. Open the desired task.
For information on opening a task, see [Editing a replication task \(page 125\)](#).
2. Click the **Task Settings** toolbar button.
The **<Task Name> - Settings** dialog box opens, displaying the following tabs:
 - *Metadata (page 1166)*
 - *Target metadata (page 1166)*
 - *Control tables (page 1169)*
 - *Bidirectional (page 1171)*
 - *Full Load (page 1171)*
 - *Full Load Settings (page 1171)*
 - *Full Load Tuning (page 1174)*

- *Change Processing (page 1174)*
 - *Apply Changes Settings (page 1174)*
 - *Store Changes Settings (page 1175)*
 - *Change Processing Tuning (page 1181)*
- *Error handling (page 1186)*
 - *Error Handling settings (page 1187)*
 - *Environmental errors (page 1187)*
 - *Data Errors (page 1188)*
 - *Table Errors (page 1191)*
 - *Apply Conflicts (page 1191)*
- *Logging (page 1192)*
- *Character substitution (page 1194)*
- *File uploads (page 1195)*
- *Message Format (page 1196)*
- *Transformations and Filters (page 1201)*
- *More options (page 1282)*

Metadata

Clicking the **Metadata** tab in the **Task Settings** window reveals the following sub-tabs:

- *Target metadata (page 1166)*
- *Control tables (page 1169)*

Target metadata

Target table schema: The schema on the target to which the source tables will be replicated if you do not want to use the source table schema (or if there is no schema in the source database).



When replicating to Hadoop-based targets, the value specified in this field will be interpreted as a database name (as opposed to a schema name).

LOB handling options




For information on how to override these settings for individual tables, see *Handling LOB columns (page 1103)*.





- *During CDC or during Full Load when the **Allow unlimited LOB size** option is enabled, LOB data types are supported only in tables with a primary key or unique index.*
- *When replicating from Microsoft SQL Server, inline LOBS will always be read directly from the logs (i.e. without lookup).*

The following LOB handling options are available:

LOB handling options

Option	Description
Replicate LOB columns	<p>When this option is selected (the default), LOB columns will be replicated.</p> <p>Note that replicating LOBs may impact performance. This is especially true in the case of the large LOBs which require Replicate to perform a lookup from the source table in order to retrieve the source LOB value.</p>
Allow unlimited LOB size	<p>Select this option - also known as Full LOB mode - to ensure that all LOBs are replicated without being truncated. This option should be selected when all (or nearly all) of the LOBs you wish to replicate are large (i.e. exceed 1 GB).</p> <div data-bbox="400 656 1386 826" style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> Note If the task's Change Processing Mode is set to "Batch optimized apply" (the default), Replicate will switch to "Transactional apply" mode to apply tables with LOBs.</p> </div>
Optimize handling when LOB size is less than (KB)	<p>Select this option when you need to replicate both small and large LOBs, and most of the LOBs are small.</p> <div data-bbox="400 949 1386 1227" style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> This option is supported with the following endpoints only:</p> <ul style="list-style-type: none"> • Sources: Oracle, Microsoft SQL server, MySQL, PostgreSQL, IBM DB2 for LUW, and Sybase ASE. • Targets: Oracle, Microsoft SQL Server, MySQL, PostgreSQL, IBM DB2 for z/OS, and Sybase ASE. </div> <p>When this option is selected, during Full Load, the small LOBs will be replicated "inline" (which is more efficient), and the large LOBs will be replicated by performing a lookup from the source table.</p> <p>During Change Processing, however, both small and large LOBs will be replicated by performing a lookup from the source table.</p> <div data-bbox="400 1491 1386 1758" style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> When this option is selected, Replicate will check all of the LOB sizes to determine which ones to transfer "inline". LOBs larger than the specified size will be replicated using Full LOB mode.</p> <p>Therefore, if you know that most of the LOBs are larger than the specified setting, it is better to use the Allow unlimited LOB size option instead.</p> </div>

Option	Description
Chunk size (KB)	<p>Optionally, change the size of the LOB chunks to use when replicating the data to the target. The default chunk size should suffice in most cases, but if you encounter performance issues, adjusting the size may improve performance.</p> <div data-bbox="400 427 1390 600" style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;">  <p><i>With some databases, data type validation occurs when the data is inserted or updated. In such cases, replication of structured data types (e.g. XML, JSON, GEOGRAPHY, etc.) may fail if the data is bigger than the specified chunk size.</i></p> </div>
Limit LOB size to (KB)	<p>Select this option if you only need to replicate small LOBs or if the target endpoint does not support unlimited LOB size. The maximum permitted value for this field is 102400 KB (100 MB).</p> <p>When replicating small LOBs, this option is more efficient than the Allow unlimited LOB size option since the LOBs are replicated "inline" as opposed to via "lookup" from the source. During Change Processing, small LOBs are usually replicated via "lookup" from the source.</p> <p>As the value of the Limit LOB size to is in bytes, the size should be calculated according to the following formulas:</p> <ul style="list-style-type: none"> • BLOB – The length of the largest LOB. • NCLOB – The length of the longest TEXT <i>in characters</i> multiplied by two (as each character is handled as a double-byte). If the data includes 4-byte characters, multiply it by four. • CLOB – The length of the longest TEXT in characters (as each character is handled as a UTF8 character). If the data includes 4-byte characters, multiply it by two. <div data-bbox="400 1330 1390 1594" style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;">  <ul style="list-style-type: none"> • <i>Any LOBs larger than the specified size will be truncated.</i> • <i>During Change Processing from Oracle source, inline BLOBs are replicated inline.</i> • <i>Changes to this setting will only affect existing tables after they are reloaded.</i> </div>



In some scenarios, tasks configured to replicate tables with multiple LOB columns may consume a large amount of memory. This is because Replicate allocates memory by multiplying the **Limit LOB size to** value by the [Commit rate during full load](#) value, the sum of which, it multiplies by the number of LOB columns being replicated. So, for example, if LOB size is limited to 5 MB and the default commit rate is used (10000 events), a task replicating 6 LOB columns will consume 300 GB of memory. Note that other factors such as the database type and version may also affect memory consumption.

Should you encounter memory consumption issues and suspect that a combination of the above factors may be the cause, stop the task and lower the value in the [Commit rate during full load](#) field. Then resume the task. Repeat this process until acceptable performance/memory levels are reached.

These instructions apply to Change Processing and Full Load tasks.



Changes to a column's LOB size while a task is running will not be reflected in the Change Table, unless the target tables are created by Qlik Replicate. In such cases, the task must be configured to drop and create the Change Table (the default) and the target tables need to be reloaded (after the LOB size has changed).

For more information on the Change Table, see *Store Changes Settings* (page 1175). For information on reloading target tables, see *Working with tasks at runtime* (page 1203).

Control tables

Control Tables provide information about the replication task as well as useful statistics that can be used to plan and manage both the current replication task and future replication tasks. Aside from the **Apply Exceptions** table which is always created, you can [choose which Control Tables to create on the target](#).

Create target control tables in schema: Enter the endpoint schema for the target Control Tables. If you do not enter any information in this field, then the tables will be created in the default location in the endpoint.



When this field is left empty, the target endpoint is MySQL, and the Multiple Endpoints option is enabled, a default database named `attrep_control` will be created on the MySQL server. The selected control tables will be created in this database.

For more information on the Multiple Endpoints option, see *Setting general connection properties* (page 951).



When replicating to a Hadoop target endpoint, the value specified in this field will be interpreted as a database name (as opposed to a schema name).

Create target control tables in tablespace: When the target endpoint is Oracle, specify the tablespace where you want the target control tables to be created. If you do not enter any information in this field, the tables will be created in the default tablespace in the target database.

Create target control table indexes in tablespace: When the target endpoint is Oracle, specify the tablespace where you want the control table indexes to be created. If you do not enter any information in this field, the indexes will be created in the same tablespace as the control tables.

Replication history time slot (minutes): The length of each time slot in the Replication History table. The default is 5 minutes.

Table selection

In addition to the **Apply Exceptions** table (required), select which of the following Control Tables you want Qlik Replicate to create on the target endpoint:

- **Replication Status:** Provides details about the current task including task status, amount of memory consumed by the task, number of changes not yet applied to the target and the position in the source endpoint from which Qlik Replicate is currently reading.
- **Suspended Tables:** Provides a list of suspended tables as well as the reason they were suspended.
- **Replication History:** Provides information about the replication history including the number and volume of records processed during a replication task, latency at the end of a CDC task, among others.
- **Change Data Partitions:** The **attrep_cdc_partitions** table contains records of partitions created on the target database when *Change Data Partitioning* (page 1177) is enabled for a Replicate task. You can use this information to identify partitioned data that needs to be further processed.
- **DDL History:** The **attrep_ddl_history** table contains a history of all supported DDL changes that occurred during a task.

For a list of DDL changes supported by Replicate, see *Supported DDL statements* (page 83). Note that DDL changes written to this Control Table are also subject to the limitations described in the section in [Limitations when Capturing DDL Changes](#).



The DDL History table is currently supported with the following target endpoints only:

- Amazon EMR
- Cloudera Data Platform (CDP) Private Cloud
- Confluent Cloud
- Databricks (Cloud Storage)
- Databricks Lakehouse (Delta)
- Google Cloud Pub/Sub
- Google Dataproc
- Hadoop
- Hortonworks Data Platform
- Microsoft Azure Databricks
- Microsoft Azure HDInsight
- Snowflake on Google

For a detailed description of these tables, see *Control Tables* (page 1298).

Bidirectional

This tab is only applicable to bidirectional replication tasks. When you click **Bidirectional** in the **Task Settings** dialog box, the **Loopback Prevention** tab is displayed. In bidirectional replication, loopback prevention is a mechanism that prevents the same data from being replicated back and forth in an endless loop. To enable loopback prevention, you need to specify a source and target **Loopback prevention table schema**.

Bidirectional replication consists of two separate tasks: Task 1 captures changes made to Endpoint A and replicates them to Endpoint B. Task 2 captures changes made to Endpoint B and replicates them to Endpoint A. When configuring Task 1 of a bidirectional replication setup, the source loopback prevention table schema must be identical to the target loopback prevention table schema specified in the **Loopback Prevention** settings of Task 2.

Likewise, when configuring Task 2 of a bidirectional replication setup, the source loopback prevention table schema must be identical to the target loopback prevention table schema specified in the **Loopback Prevention** settings of Task 1.



*Oracle schemas are case-sensitive. Therefore, when specifying an Oracle table schema, make sure to use the correct case in the **Loopback Prevention** settings in both Tasks.*

For instructions on setting up bidirectional replication, see *Bidirectional replication (page 111)*.

Full Load

Clicking the **Full Load** tab in the **Task Settings** window reveals the following sub-tabs:

- *Full Load Settings (page 1171)*
- *Full Load Tuning (page 1174)*

Full Load Settings

Click the **Full Load Settings** sub-tab to configure the following:

Full Load Processing is ON/OFF.

Click this button to toggle full load on or off. The initial setting is determined when *Adding tasks (page 110)*.

When full load is ON, Qlik Replicate loads the initial source data to the target endpoint.



Full load can be turned on or off at any stage even if change processing is on. Once the task begins to process changes, the full load on/off switch is used only as additional protection against accidental or unauthorized reload.

Target table preparation

If target table already exists: Select one of the following from the list to determine how you want to handle loading the target at full-load start up:



The option to drop or truncate the target tables is relevant only if such operations are supported by the source endpoint.

- **DROP and Create table:** The table is dropped and a new table is created in its place.



Replicate Control Tables will not be dropped. However, any suspended tables that are dropped will also be deleted from the [attrep_suspended_tables](#) Control Table if the associated task is reloaded.

- **TRUNCATE before loading:** Data is truncated without affecting the table metadata. Note that when this option is selected, enabling the [Create primary key or unique index after full load completes](#) option will have no effect.
- **ARCHIVE and CREATE table:** A copy of the existing table will be saved to the same schema before the new table is created. The archived table name will be appended with a timestamp, indicating when the archiving operation occurred (e.g. customers_20170605175601).



Currently this option is only available for the [Hadoop target endpoint](#).

- **Do nothing:** Existing data and metadata of the target table will not be affected. New data will be added to the table.



Replicate expects the source column data types to be compatible with the corresponding target column data types. If you choose either **TRUNCATE before loading** or **Do nothing** and one or more target data types are different than the data types for the corresponding source columns, use a transformation to convert the data types as required.

For information on creating data type transformations, see *Defining transformations for a single table/view* (page 1084).

Primary Key or Unique Index Creation

Create primary key or unique index after full load completes: Select this option if you want to delay primary key or unique index creation on the target until after full load completes.

Stopping the Task after Full Load

After Full Load completes, stop the task: You can set the task to stop automatically after Full Load completes. This is useful if you need to perform DBA operations on the target tables before the task's Apply Changes (i.e. CDC) phase begins.

During Full Load, any DML operations executed on the source tables are cached. When Full Load completes, the cached changes are automatically applied to the target tables (as long as the **Before/After cached changes have been applied** option(s) described below are disabled).



This feature is not available for bidirectional replication tasks.

Select **Before cached changes have been applied** to stop the task before the cached changes are applied and/or **After cached changes have been applied** to stop the task after the cached changes are applied.

Selecting the **Before cached changes have been applied** option will stop the task after Full Load completes. Selecting the **After cached changes have been applied** option will stop the task as soon as data is consistent across all tables in the task.



When configuring Replicate to stop the task after Full Load completes, note the following:

- The task does not stop the moment Full Load completes. It will be stopped only after the first batch of changes has been captured (as this is what triggers the task to stop). This might take a while depending on how frequently the source database is updated. After the task stops, the changes will not be applied to the target until the task is resumed.
- The task will stop after Full Load completes, even if there are no cached changes to apply.
- The **After cached changes have been applied** option is not supported with all file-based and Hadoop-based target endpoints, namely:
 - **File-based:** File, Amazon S3, Microsoft Azure ADLS, and Google Storage.
 - **Hadoop-based:** Hadoop, Hortonworks Data Platform, Amazon EMR, Microsoft Azure HDInsight, Google Dataproc, Cloudera Data Platform (CDP) Private Cloud, and Microsoft Azure Databricks.
- Choosing to stop the task before cached changes have been applied may adversely affect performance, since the cached changes will only be applied to tables (even those that have already completed Full Load) after the last table completes Full Load.
- When the **Before/After cached changes have been applied** option is selected and a DDL is executed on one of the source tables during the Full Load process (in a Full Load and Apply Changes task), Replicate will reload the table. This effectively means that any DML operations executed on the source tables will be replicated to the target before the task stops.
- When working with the File Channel endpoint, these options should be set in the remote File Channel task and not in the local File Channel task.
For more information on the File Channel endpoint, see *Using the Qlik Replicate file channel* (page 1061).

Duplicate Record Prevention



Supported when using the IBM DB2 for z/OS and IBM DB2 for iSeries source endpoints only.

Select the **Eliminate creation of duplicate records on full load** option if you need to prevent duplicate records from being replicated during Full Load. You can either set the option at task level or per table.

Note that selecting this option could impact performance as Replicate instructs the source database to return the table records by Primary Key order and then removes any duplicate records.

For information on preventing creation of duplicate records at table level, see *Full Load* (page 1106).

Full Load Tuning

Click the **Full Load Tuning** sub-tab to configure the following:

Tuning settings:

- **Maximum number of tables to load in parallel:** Enter the maximum number of tables to load into the target at one time. The default value is **5**.
- **Transaction consistency timeout (seconds):** Enter the number of seconds that Qlik Replicate waits for transactions to close, if they are open when the task starts, before beginning the Full Load operation. The default value is **600** (10 minutes). Qlik Replicate will begin the full load after the timeout value is reached even if there are open transactions.



To replicate transactions that were open when Full Load started but were only committed after the timeout value was reached, you need to reload the target tables.

- **Commit rate during full load:** The maximum number of events that can be transferred together. The default value is **10000**.

Change Processing

Clicking the **Change Processing** tab in the **Task Settings** window reveals the following sub-tabs:

- *Apply Changes Settings (page 1174)*
- *Store Changes Settings (page 1175)*
- *Change Processing Tuning (page 1181)*

Apply Changes Settings

Click the **Apply Changes Settings** sub-tab to configure the following:

Apply Changes is ON/OFF:

Click this button to toggle Apply Changes (Change Processing) on or off. The initial setting is determined when *Adding tasks (page 110)*.

When Apply Changes is ON, Qlik Replicate processes the changes. You can view the change processing in the Monitor. For more information, see *Monitoring change processing operations (page 1224)*.



When you turn on apply changes you must reload the task or position back to the point of the previous reload.

DDL handling policy: Determine how to handle the target table for the change capture:



- Executing a DDL on a source table during the Full Load process in a Full Load and Apply Changes task will cause Replicate to reload the table.
- The option to drop or truncate the target tables is relevant only if such operations are supported by the source endpoint.
- Executing the Rename Table DDL with "Ignore ALTER" selected will cause events to continue being delivered until you stop and then resume the task.

- When source table is dropped, select one of the following:
 - **DROP target table**
 - **Ignore DROP**
- When source table is truncated, select one of the following:
 - **TRUNCATE target table**
 - **Ignore TRUNCATE**
- When source table is altered, select one of the following:
 - **ALTER target table**
 - **Ignore ALTER**

Store Changes Settings

When **Store Changes** is turned on in the task settings, you can configure the **Store Changes Settings** for a replication task.

Store changes processing is ON/OFF:

Click this button to toggle **Store Changes** on or off. The initial setting is determined when *Adding tasks* (page 110). If this option is ON, changes are stored in either Change Tables or an Audit Table.

For more information about storing and applying changes, see *Using an Audit table* (page 1289) and *Working with Change Tables* (page 1283).



Store Changes can be turned on or off at any time without affecting anything in the task. Changes that are processed and not stored as a result of change storage being turned off can be recovered only by setting the task to an earlier point in time.

If Store Changes is **ON**, use the following options to determine how to store changes. Changes can be stored in Change Tables or in a single Audit table. From the **Store changes in** drop-down list, choose either **Change tables** or **Audit table** according to your needs.

Storing changes in Change Tables

The following section describes the options that are available when storing changes in Change Tables.

- **Suffix:** Type a string to use as the suffix for all Change Tables. The default value is __ct. The Change Table names are the name of the target table with the suffix appended. For example, if you have a table called HR and use the default value, the name of the Change Table will be HR__ct.

For more information, see *Working with Change Tables* (page 1283).

- **Header column prefix:** Type a string to use as the prefix for all of the Change Table header columns. The default value is `header__`.
For example, the header column `stream_position` when using the default value is called `header__stream_position`.
For more information, see *Change Tables* (page 1284).
- **DDL options:** Select one of the following options to determine how to handle DDL operations on the source tables:
 - **Apply to change table** - DDL operations to the source tables (such as a column being added) will be applied to the corresponding Replicate Change Tables only.
 - **Ignore** - All DDL operations to source tables will be ignored.
 - **Apply to change table and target table**
This is the default option. When this option is selected, DDL operations to the source tables will be applied both to the Replicate Change Tables and to the corresponding target tables.
- **On UPDATE:** Select one of the following options to determine how to store UPDATES to the source tables:
 - **Store before and after image** - To store both the pre-UPDATE data and the post-UPDATE data.
 - **Store after image only** - To store only the post-UPDATE data.

Change table creation:

If Change Table exists when full load starts: Select one of the following to determine how you want to handle loading the Change Tables when Full Load replication starts:

- **DROP and CREATE Change Table:** The table is dropped and a new table is created in its place.
- **ARCHIVE and CREATE Change Table:** A copy of the existing table will be saved to the same schema before the new table is created. The archived table name will be appended with a timestamp, indicating when the archiving operation occurred (e.g. `Customers__ct_20170605175601`).



Currently this option is only available for the [Hadoop target endpoint](#).

- **Delete old changes and store new changes in existing Change Table:** Data is truncated and added without affecting the table metadata.
- **Keep old changes and store new changes in existing Change Table:** Data and metadata of the existing Change table are not affected.

Change Data Partitioning



This feature is supported with the following target endpoints only:

- *Hadoop (Hortonworks, Cloudera, and MapR)*
- *File*
- *Amazon S3*
- *Databricks (Cloud Storage)*
- *Microsoft Azure HDInsight*
- *Google Cloud Storage*
- *Microsoft Azure ADLS*
- *Cloudera Data Platform (CDP) Private Cloud*
- *Google Data Proc*
- *Amazon EMR*
- *Hortonworks Data Platform (HDP)*

In a standard replication task, changes are replicated to the target in no particular order. Change Data Partitioning enables processing of Change Data from many tables in a consistent fashion. You can define the duration of partitions as well as the partitioning base time, thereby ensuring overall consistency of the partitioned data (i.e. no partial transactions, no order headers without order lines, and so on.)

The partitioned data is stored in the Replicate Change Tables. When the [Change Data Partitions](#) table is selected (in the *Control tables* (page 1169) tab), information about the partitions will be recorded in the [attrep_cdc_partitions](#) Control Table on the target database. This information can be used to identify partitioned data that needs to be further processed.

The partitioning options are as follows:

- **Off** - Replicate Change Data without partitioning.
- **Partition every** - Specify the length (in hours and minutes) of each partition.



It is recommended to specify a partition length in excess of one hour. Although specifying a partition length less than one hour may improve latency, creating many partitions on the target may also impact (target) performance (especially in systems with large volumes of changes).

If you resume a task from BEFORE the time that the last partition was created, Replicate will write to a partition that has already been closed.

- **Partition base time** - Partitions are created during a 24 hour time period, which is calculated according to the specified “Partitioning base time” on the source database (in UTC time). For example, a partition interval of 8 hours with a “Partitioning base time” time of 02:00 will create the following partitions: 02:00-10:00, 10:00-18:00, 18:00-02:00 - but not necessarily in that order. For instance, if a task started at 01:00, then the timeframe of the first partition will be 18:00-02:00. Additionally, if a task

started in the middle of a partition (e.g. at 04:00), its Change Data will be inserted into the 02:00-10:00 partition (even though no changes were captured before 04:00).

- **Speed partition mode**



This feature is supported with Hadoop-based target endpoints only.

By default, Change Data Partitions for all tables in a replication task are registered on the target at the same time. As soon as the partitions are registered, information about them (such as a partition's start and end time) is also published to the [Change Data Partitions](#) Control Table. This ensures data consistency across all of the replicated tables.

In **Speed partition mode**, rather than waiting for all Change Data files to be uploaded (for all tables), Replicate creates and registers a partition per-table as soon as the partition's first data file is uploaded (to its table). Creating and registering partitions *before* all the Change Data has been uploaded is especially useful for consuming applications such as Qlik Compose that need to process changes with a minimum of delay, as the Change Data rapidly becomes available for consumption, even if it is incomplete.

For example, if an endpoint is configured to upload Change Data files every five minutes, each batch of changes can be processed immediately, even if the Partition every interval is set to six hours. However, despite significantly reducing latency, working in this mode may have some possible drawbacks:

- Accumulation of a large number of files per partition, which may impact performance on the target.
It is therefore strongly recommended to clean up old partitions using the Partition Retention options (see below) provided for this purpose.
- Inconsistent results may be encountered when querying several tables due to the Change Data arriving on the target at different times.



Change Tables that were created before Change Data Partitioning was enabled need to be dropped or renamed so that they can be recreated with the [additional "partition_name" column](#).

Deleting old partitions periodically

Over time, Change Data Partitioning can result in a large number of files and partitions accumulating on the target system, which may significantly impact performance. Therefore, to ensure optimal performance, best practice is to delete old partitions from time to time.

There are two ways of deleting processed partitions: periodically and ad-hoc. This topic explains how to set up periodic partition deletion. For information on ad-hoc partition deletion, see *Deleting old partitions manually* (page 1237).



Currently, this feature is supported with the Microsoft Azure Databricks endpoint only.

Setting up periodic deletion is a two-phase process which involves specifying a deletion interval in the UI, and setting a partition retention barrier using the Enterprise Manager API.

To do this:

1. Enable the **Partition Retention** option.
Any partitions created while this option is disabled will not be deleted when deletion is initiated (either periodic or manual). Therefore, the **Partition Retention** option should only be disabled if you are absolutely certain that periodic deletion will not be initiated at any point in the future.
2. Specify a partition deletion interval in **Day, Hours or Minutes**.
To prevent the deletion of open partitions or recently closed partitions, the partition deletion interval must be at least double the **Partition every** value.
3. Set a retention barrier date by calling the relevant Enterprise Manager API method.
To ensure that only old (i.e. processed) partitions will be deleted, periodic deletion requires a retention barrier date to be set. The retention barrier date should be set by the consuming application each time it finishes processing a partition. A consuming application can be any application (Qlik or otherwise) tasked with processing the partitioned data. There may be several consuming applications, each of which sets its own retention barrier. In this case, partitions will be deleted up to the earliest retention barrier. For example, if Application A sets July 7th, 2020 as a barrier, Application B sets August 7th, 2020 as a barrier, and Application C sets September 7th, 2020 as a barrier, partitions will be deleted up to July 7th, 2020.
As soon as a retention barrier is set, periodic deletion will start to run according to the defined interval, until any of the following occurs:
 - The **Partition Retention** option is disabled
 - An API call removes the retention barrier
 - All partitions until the retention barrier have been deleted

For more information on setting and removing a partition retention barrier using the API, refer to the Enterprise Manager Developer's Guide.

Limitations and considerations

- If a connection error occurs during the partition deletion operation, the operation will fail (with a warning). The operation will continue from the point of failure the next time that deletion is performed (either periodically or manually).
- If a specific object (partition or storage) cannot be deleted due to other errors (i.e. not connection errors), a warning will be written to the log and deletion of the remaining objects will continue. Partitions that cannot be deleted due to such errors must be deleted directly from the database by the DBA.
- To prevent deletion of open partitions, the last partition in a table will not be deleted, even if meets the criteria for deletion.
- Deletion of partitions in renamed tables is not supported.
- Reloading a target with the [Drop and Create tables](#) option (the default) will delete both the tables and their partitions, regardless of any partition deletion policy. However, reloading a target with the **Do nothing** option, will leave the partitions untouched. In this case, partitions created prior to the target reload cannot be deleted using the Replicate partition deletion tools.

Selecting Change Table header columns

The Change Table header columns provide information about the Change Processing operation such as the type of operation (e.g. INSERT), the commit time, and so on. If you do not need this information, you can configure Replicate to create the Change Tables without some or all of the header columns, thereby reducing their footprint in the target database. To do this, clear the check boxes next to the header columns that you wish to exclude.

Note that you cannot remove additional columns or restore columns while a task is running. To change your initial selection, you first need to stop the task, then modify your selection, and finally reload the target tables.



When [Change Data Partitioning](#) is enabled, an extra header column named "partition_name" is added to the Change Tables and automatically selected in the UI. As this column is required, it cannot be excluded.

For a description of the header columns, see *Change Tables* (page 1284).

Storing changes in an Audit table

The following section describes the options that are available for storing changes in an Audit table.



- If your source tables contain records exceeding 4096 characters, to prevent truncation of these records, you must turn on the Task Settings' [Replicate LOB columns](#) option (even if the source tables do not contain any LOB columns).
- LOB columns with unlimited size are not supported in the CHANGE_RECORD and BU_CHANGE_RECORD columns. The other fields will be recorded but the LOB will have a NULL value.
- For a description of the audit table structure, see *Using an Audit table* (page 1289).

- **Audit table schema:** Specify a schema if you do not want the Audit table to be created under the target endpoint's default schema.

The default schema are as follows:

Default schema by endpoint

Endpoint	Default Schema
Pivotal Greenplum	Public
Amazon Redshift	Public
Oracle	The connected user's user name.
Teradata	The endpoint name.
All others	The user's default schema.

- **Audit table tablespace:** This option is only available when the task's target endpoint is Oracle. Enter the tablespace name on the target where you want the Audit table to be created. If you do not enter any information in this field, then the tables will be created in the default permanent tablespace.
- **Audit table name:** Specify a name for the Audit table.
The default value is `attrep__audit_table`.

Audit table creation:

If audit table exists when the target is reloaded: Select one of the following to determine how you want to handle the Audit table when the target is reloaded:

- **DROP and CREATE audit table:** The Audit table is dropped and a new table is created in its place.
- **ARCHIVE and CREATE audit table:** A copy of the existing table will be saved to the same schema before the new table is created. The archived table name will be appended with a timestamp, indicating when the archiving operation occurred (e.g. `attrep_audit_table_20170605175601`).



Currently this option is only available for the [Hadoop target endpoint](#).

- **Delete old changes and store new changes in existing audit table:** Data is truncated and added without affecting the Audit table metadata.
- **Keep old changes and store new changes in existing audit table:** Data and metadata of the existing Audit table are not affected.

For a description of the audit table structure, see *Using an Audit table* (page 1289).

Change Processing Tuning

Click the **Change Processing Tuning** sub-tab to fine-tune the Apply Changes settings.

Change Processing Mode

Determine which method will be used to apply changes.




Changes to tables without a Unique Index or Primary Key will always be applied in **Transactional apply mode**.

- **Transactional apply:** Select this to apply each transaction individually, in the order it is committed. In this case, strict referential integrity is ensured for all tables.




Applying cached events in transactional mode to endpoints that do not enforce constraints (such as Vertica and IBM Netezza), may result in duplicate records on the target. This is because such endpoints do not return duplicate errors.


- **Batch optimized apply:** Select this to commit the changes in batches. In this case, a pre-processing action occurs to group the transactions into batches in the most efficient way. This may affect transactional integrity. Therefore, you must select one of the following to determine how the system will handle referential integrity issues:

 In the event of a recoverable error during the Batch optimized apply process, multiple `attrep_changes` (Net Changes) tables might be created in the target database (each with a unique name). These tables will need to be deleted manually as the automatic cleanup process will not delete them.

- Preserve transactional integrity

 This option is only supported when replicating to an Oracle target.

- Allow temporary lapses in transactional integrity to improve performance

 These options are not displayed in bidirectional tasks since such tasks always use the "Preserve transactional integrity" option.

The following target endpoints do not support applying binary data types in Batch optimized apply mode:

ODBC, SAP Sybase IQ, SAP Sybase ASE, Vertica, IBM Netezza, Teradata, and Amazon Redshift.

When LOB columns are included in the replication, **Batch optimized apply** can only be used with the **Limit LOB size to** option. For more information about including LOB columns in the replication, see *Metadata* (page 1166).

Using Batch optimized apply to apply changes to tables with foreign keys is not supported.

Batch tuning

The following options are available when **Batch optimized apply** is selected as the **Change Processing Mode**:

- **Apply batched changes in intervals:**
 - **Longer than:** The minimum amount of time to wait between each application of batch changes. The default value is **1**.
Increasing the **Longer than** value decreases the frequency with which changes are applied to the target while increasing the size of the batches. This can improve performance when applying changes to target endpoints that are optimized for processing large batches, such as Teradata, Vertica, and Pivotal Greenplum.
 - **But less than:** The maximum amount of time to wait between each application of batch changes (before declaring a timeout). In other words, the maximum acceptable latency. The default value is **30**. This value determines the maximum amount of time to wait before applying the changes, after the **Longer than** value has been reached.
- **Force apply a batch when processing memory exceeds (MB):** The maximum amount of memory to use for pre-processing in **Batch optimized apply mode**. The default value is **500**.
For maximum batch size, set this value to the highest amount of memory you can allocate to Qlik Replicate. This can improve performance when applying changes to target endpoints that are optimized for processing large batches, such as Teradata, Vertica, and Pivotal Greenplum.

- **Apply batched changes to multiple tables concurrently:** Selecting this option should improve performance when applying changes from multiple source tables.
 - **Maximum number of tables:** The maximum number of tables to apply batched changes to concurrently. The default is five, the maximum is 50, and the minimum is two.



Limitations:

When the **Apply batched changes to multiple tables concurrently** option is enabled, the following limitations apply:

- **Supported by the following target endpoints only:** Snowflake on Azure, Snowflake on AWS, Snowflake on Google, Microsoft SQL Server, Amazon Redshift, Microsoft Azure Databricks Delta, Microsoft Azure Synapse Analytics, and Microsoft Azure SQL Database.

- **Error handling limitations:**

The global error handling policy will be unavailable.

The task error handling policy defaults will be unchanged for **Environmental** and **Table** errors, but the defaults for **Data** errors and **Apply Conflicts** errors will be as follows:

Data Errors:

- **Data truncation errors:** Ignore record
- **Other data errors:** Suspend table

Apply Conflicts Errors:

- **Deletes:** Ignore record
- **Inserts:** UPDATE the existing target record
Note that this is not relevant for a Snowflake target (as Snowflake does not support Primary Keys).
- **Updates:** Ignore record

Escalation Action:

- The **Escalation action** for both **Data** errors and **Apply Conflicts** is not supported.
- **Control Table limitations:**
 - The [attrep_apply_exception Control Table](#) is not supported.

- **Limit the number of changes applied per change processing statement to:** To limit the number of changes applied in a single change processing statement, select this check box and then optionally change the default value. The default value is **10,000**.



This option is not available for tasks configured with the following target endpoints:

- Microsoft SQL Server
- Microsoft Azure SQL Database
- Databricks Lakehouse (Delta) (When using Microsoft Azure Data Lake Storage (ADLS) Gen2 storage)
- Microsoft Azure Synapse Analytics
- Google Cloud SQL for SQL Server
- Amazon Redshift
- Snowflake on AWS
- Snowflake on Azure
- Snowflake on Google

The following options are available when **Transactional apply** is selected as the **Change Processing Mode**:

- **Minimum number of changes per transaction:** The minimum number of changes to include in each transaction. The default value is **1000**.



*Replicate applies the changes to the target either when the number of changes is equal to or greater than the **Minimum number of changes per transaction** value OR when the batch timeout value is reached (see below) - whichever occurs first. Because the frequency of changes applied to the target is controlled by these two parameters, changes to the source records may not immediately be reflected in the target records.*

- **Maximum time to batch transactions before applying (seconds):** The maximum time to collect transactions in batches before declaring a timeout. The default value is **60**.

Transaction offload tuning

The following tuning options are available, regardless of which **Change processing mode** is selected:

- **Offload transaction in progress to disk if:**
Qlik Replicate usually keeps transaction data in memory until it is fully committed to the source and/or target. However, transactions that are larger than the allocated memory or that are not committed within the specified time limit will be offloaded to disk.
 - **Transaction memory size exceeds (MB):** The maximum size that all transactions can occupy in memory before being offloaded to disk. The default value is **1000**.
 - **Transaction duration exceeds (seconds):** The maximum time that each transaction can stay in memory before being offloaded to disk. The duration is calculated from the time that Qlik Replicate started capturing the transaction. The default value is **60**.

Miscellaneous tuning

- **Statements cache size (number of statements):** The maximum number of prepared statements to store on the server for later execution (when applying changes to the target). The default is 50. The maximum is 200.

- **Store task recovery data in target database:** Select this option to store task-specific recovery information in the target database. When this option is selected, Replicate creates a table named `attnetrep_txn_state` in the target database. This table contains transaction data that can be used to recover a task in the event that the files in the **Data** folder are corrupted or if the storage device containing the **Data** folder has failed.
For more information about this option, see *Recovering from data folder loss or corruption* (page 1210).
- **DELETE and INSERT when updating a primary key column:** For tasks configured with streaming target endpoints (for example, Kafka), the DELETE+INSERT statements will be executed on the actual target. For all other target endpoints, they will be executed in the associated Change Tables. This option requires full supplemental logging to be turned on in the source database.
- **Apply changes using SQL MERGE** - When this option is *not* selected, the Batch optimized apply operation executes separate bulk INSERT, UPDATE and DELETE statements for each of the different change types in the Replicate Net Changes table.



*The Replicate Net Changes table is created on the target when working in Batch optimized apply mode and contains records that were changed on the source. It is truncated each time the source changes are applied to the target. For more information on the Net Changes table, see *Net Changes table* (page 34)*

While this method is highly efficient, enabling the **Apply changes using SQL MERGE** option is even more efficient when working with endpoints that support this option.

This is due to the following reasons:

- It reduces the number of SQL statements run per table from three to one. Most UPDATE operations in large, immutable, file-based cloud databases (such as Google Cloud BigQuery), involve rewriting of affected files. With such operations, the reduction of per-table SQL statements from three to one is very significant.
- The target database only needs to scan the Replicate net changes table once, significantly reducing I/O.
 - **Optimize inserts:** When **Apply changes using SQL MERGE** is selected together with this option and the changes consist of INSERTs only, Replicate will perform INSERTs instead of using SQL MERGE. Note that while this will usually improve performance and thereby reduce costs, it might also result in duplicate records in the target database.



- The **Apply changes using SQL MERGE** and **Optimize inserts** options are available for tasks configured with the following target endpoints only:
 - Google Cloud BigQuery
 - Databricks Lakehouse (Delta)
 - Snowflake on Azure, Snowflake on Google, and Snowflake on Amazon
- The **Apply changes using SQL MERGE** and **Optimize inserts** options are not supported with the following source endpoints:
 - Salesforce
 - Oracle

For more information, see the "Limitations and Considerations" topic for these endpoints.

- When the **Apply changes using SQL MERGE** option is enabled (either alone or together with the **Optimize inserts** option):
 - Non-fatal data errors or data errors that cannot be recovered will be handled as table errors
 - The [Global error handling policy](#) will not be available
 - The [Apply Conflicts error handling policy](#) will be preset and read-only. If the **Optimize inserts** option is also selected, the **Duplicate key when applying INSERT** Apply Conflicts error-handling option will be set to **Allow duplicates in the target**.
 - Some of the [Data error handling policy](#) options will not be available.
 - The operations will only be performed on the final target tables. For Change Tables or the Audit Table, INSERTs will be performed.

Error handling

Qlik Replicate handles different types of errors during its operation. The way the system should respond to these errors depends on several aspects, including the component where the error occurred, the type of error, and the scope of the error. Because different sites may have different requirements for error behavior, Qlik Replicate lets you configure the error handling.

You can also add an environment variable that instructs Replicate to create dump files in the event of a crash. The dump files can then be used by Qlik Support to troubleshoot the cause of the crash. For more information, see [Creating Dump Files](#).

You can determine whether or not to override the global error handling settings. For more information, see [Error Handling Settings](#).



The option to set a Global Error Handling policy is not available when the **Apply batched changes to multiple tables concurrently** option is selected in the Task Settings' [Change Processing Tuning](#) tab.

Clicking the **Error Handling** tab in the **Task Settings** window reveals the following sub-tabs:

- **Environmental Errors:** An error that is caused by an environmental problem in the source or target endpoint or on the network. Some examples of environmental errors are loss of communication with the source or target endpoint, restarting a database, or network problems.
- **Data Errors:** An error related to data processing at the record level. Some examples of data errors are conversion errors, errors in transformations, or bad data.
- **Table Errors:** An error in processing data or metadata for a specific table. This only includes general table data and not an error that relates to a specific record.
- **Apply Conflicts:** Errors that occur when the target endpoint is not synchronized with the source endpoint when processing changes.
This can cause duplicate key errors on INSERT operations or zero rows affected on UPDATE/DELETE operations.

Error Handling settings

The option to switch between the Global Error Handling policy and a Task Error Handling policy is available in each of the **Error Handling** sub-tabs. However, the policy you enable will be applied to all error types, regardless of where it was enabled. For example, you cannot enable a Task Error Handling policy for Data Errors and then enable the Global Error Handling policy for Table Errors and Environmental Errors.

For information on setting the global error handling policy, see *Global error handling (page 1264)*.

To set a Task-Specific Error Handling policy:

- Click the **Change to Task Policy** button in any of the **Error Handling** sub-tabs.

To revert to the Global Error Handling policy:

1. Click the **Change to Global Policy** button in any of the **Error Handling** sub-tabs.
2. Click **OK** when prompted to confirm your action.

Environmental errors

Click the **Environmental Errors** sub-tab and then click [Change to Task Policy](#) to configure the following:

- **Maximum retry count:** Select this option and then specify the maximum number of attempts to retry a task when a recoverable environmental error occurs.
 1. To never retry a task, clear the check box or specify "0".
 2. To retry the task an infinite number of times, specify "-1" (the global error handling default).When the system attempts to retry the task the designated number of times, the task is stopped and manual intervention is required.
- **Interval between retry attempts:** Use the counter to select or type the number of seconds that the system waits between attempts to retry a task.
Valid values are 0-2,000.
- **Increase retry interval for long outages:** Select this check box to increase the retry interval for long outages. When this option is enabled, Replicate doubles the interval between each retry attempt and the next, until the **Maximum retry interval** is reached (and continues retrying according to the specified maximum interval).

- **Maximum retry interval:** Use the counter to select or type the number of seconds to wait between attempts to retry a task when the **Increase retry interval for long outages** option is enabled. Valid values are 0-2,000.

Data Errors

Click the **Data Errors** sub-tab and then click [Change to Task Policy](#).



Data error handling is not relevant for Full Load replication. It is applicable to Apply Changes (CDC) replication only.

Data truncation errors

- **For a data truncation error:** Click the triangle to open the list and select what happens when an truncation occurs in one or more specific records. You can select one of the following from the list:
 - **Ignore record:** The task continues and the error is ignored.
 - **Log record to the exceptions table** (default): The task continues and the error is written to the exceptions table.
 - **Suspend table:** The task continues, but data from the table with the error record is moved into an error state and its data is not replicated
 - **Stop task:** The task is stopped and manual intervention is required.



Data truncation error handling is only supported in the following with the following target endpoints: MySQL, PostgreSQL, Oracle, Microsoft SQL Server, SAP Sybase ASE, File, Amazon Redshift, Microsoft Azure Database for PostgreSQL, Google Cloud SQL for PostgreSQL, Microsoft Azure Database for MySQL, Google Cloud SQL for MySQL, and Microsoft Azure SQL Database

NOT NULL constraint violations

For source tables, Replicate can explicitly check for NULL values in each data column of each record, and whether the same column in the corresponding target table is marked as NOT NULL. In such cases, applying the record to the target will trigger a NOT NULL constraint violation that will be difficult to recover from if the task is running in Batch Optimized Apply mode. Therefore, the record is not applied to the target. Instead, the record is handled as an error (see below) without affecting other records in the batch being applied. While this way of handling NOT NULL constraint violation is very effective, it can impact performance, and if the target database does not actually enforce NOT NULL constraint violations, Replicate can be configured to skip that check in order to save CPU time and improve performance. Because Replicate generally knows what target databases enforce NOT NULL constraint violations, it is safe to always let Replicate activate this protection based on the target endpoint type. Still, Replicate gives you the option to override this default and either force a check (with its slight overhead) or disable the check and risk NOT NULL constraint violations.



When a batch update fails due to a NOT NULL constraint violation, Replicate has no way of knowing which of the records in the batch caused that violation. Consequently, Replicate switches to a "one-by-one" strategy of applying changes. While this strategy is acceptable for OLTP databases (although even with those databases, it is much slower than Batch optimized apply), it is not practical for analytical data warehouses, specifically those that use columnar storage in files or micro-partitions (for example, Snowflake, Azure Synapse Analytics, Google BigQuery, and Databricks). As such databases are not designed for one-by-one record updates, performing such updates might take an unreasonable amount of time and consume excessive resources. Moreover, if the target database is a cloud data warehouse that charges based on activity, the one-by-one error handling strategy could result in extremely high costs.

For **NOT NULL constraint violations**, select one of the following:

- **Endpoint-determined:** This means that the endpoint type (which can be an OLTP database or an analytical data warehouse) determines whether Replicate checks for NOT NULL constraint violations.



Some target endpoints do not support checking for NOT NULL constraint violations. In such cases, the tooltip next to the **Endpoint-determined** option will indicate that the current target endpoint does not allow checking for NOT NULL constraints.

For OLTP-based target endpoints, the default is not to check for NOT NULL constraint violations as it is less critical (from a cost perspective) if a NOT NULL constraint violation occurs during the task. However, if a NOT NULL constraint violation occurs on any of the tables during the task, Replicate will switch to one-by-one mode, which will impact performance to a certain degree (depending on the number and size of the tables being replicated).

You can override the endpoint-determined handling by selecting either the **Check** or **Don't check** options described below.



The string **(Check)** or **(Don't check)** will appear after **Endpoint-determined** according to the selected target endpoint type.

- **Check:** Select this if you always want Replicate to check for NOT NULL constraint violations. This might be useful if you know beforehand that some of the source table columns contain NULL values and the corresponding target table columns are configured with NOT NULL constraints. In this case, Replicate will not switch to one-by-one mode, but will continue change processing in Batch Optimized Apply mode. If a source table that will violate a NOT NULL constraint is detected during the check, Replicate will take the action described below. The default action is to suspend the table.
- **Don't check:** Select this if you never want Replicate to check for NOT NULL constraint violations. This is recommended if you are confident that no such violations will occur during the task or if you do not mind Replicate switching to one-by-one mode if a NOT NULL constraint violation occurs.

Handling options when a NOT NULL constraint violation is discovered during a check:

For the **Endpoint-determined** and **Check** options, you can select how Replicate will handle the NOT NULL constraint violation:

- **Suspend table:** This is the default. The task continues, but tables with NULL values are suspended and their data is not replicated.
- **Ignore record:** The task continues and the error is ignored.
- **Log record to the exceptions table:** The task continues, but records with NULL values are written to the exceptions table.
- **Stop task:** The task is stopped and manual intervention is required.

Other data errors

- **For other data errors:** Click the triangle to open the list and select what happens when an error occurs in one or more specific records. You can select one of the following from the list:
 - **Ignore record:** The task continues and the error is ignored.
 - **Log record to the exceptions table** (default): The task continues and the error is written to the exceptions table.
 - **Suspend table:** The task continues, but data from the table with the error record is moved into an error state and its data is not replicated
 - **Stop task:** The task is stopped and manual intervention is required.

Escalating error handling

- **Escalate error handling when other data errors reach (per table):** Select this check box to escalate error handling when the number of non-truncation data errors (per table) reaches the specified amount. Valid values are 1-10,000.



The escalation options are not available when the [Apply changes using SQL MERGE](#) task setting is enabled.

- **Escalation action:** Choose what action Replicate should perform when error handling is escalated. Note that the available actions are dependent on the action selected from the **For other data errors** drop-down list described above.
 - **Log record to the exceptions table:** The task continues, but the record with the error is written to the exceptions table.
 - **Suspend table** (default): The task continues, but data from the table with the error record is moved into an error state and its data is not replicated.



The behavior differs according to the Change Processing Mode:

- In **Transactional apply** mode, the last changes will not be replicated
- In **Batch optimized apply** mode, a situation is possible where there will be no replication of data or data replication will occur in part

- **Stop task:** The task is stopped and manual intervention is required.

Table Errors

Click the **Table Errors** sub-tab and then click [Change to Task Policy](#) to configure the following:

When encountering a table error: Select one of the following from the drop-down list:

- **Suspend table** (default): The task continues but data from the table with the error record is moved into an error state and its data is not replicated
- **Stop task:** The task is stopped and manual intervention is required.

Escalate error handling when table errors reach (per table): Select this check box to escalate error handling when the number of table errors (per table) reaches the specified amount. Valid values are 1-10,000.

- **Escalation action:** The escalation policy for table errors is set to **Stop task** and cannot be changed.

Apply Conflicts



The Apply Conflicts error handling policy will be preset and uneditable when the [Apply changes using SQL MERGE](#) task setting is enabled.

Click the **Apply Conflicts** sub-tab and then click [Change to Task Policy](#) to configure the following:

No record found for applying a DELETE: Click the triangle to open the list and select what happens when there is a conflict with a DELETE operation. You can select one of the following from the list:

- **Ignore record** (default): The task continues and the error is ignored.
- **Log record to the exceptions table:** The task continues and the record is written to the exceptions table.
- **Suspend table:** The task continues but data from the table with the error record is moved into an error state and its data is not replicated.
- **Stop task:** The task is stopped and manual intervention is required.

Duplicate key when applying an INSERT: Click the triangle to open the list and select what happens when there is a conflict with an INSERT operation. You can select one of the following from the list:

- **Ignore record:** The task continues and the error is ignored.
- **Log record to the exceptions table** (default): The task continues and the record is written to the exceptions table.
- **Suspend table:** The task continues but data from the table with the error record is moved into an error state and its data is not replicated.
- **Stop task:** The task is stopped and manual intervention is required.
- **Update the existing target record:** The target record with the same primary key as the INSERTED source record is updated.
- **Allow duplicates in the target:** Allows duplicate primary key records to be added to the target table. When **Optimize inserts** is enabled, this is the only option that is available at task level error handling. For more information on the **Optimize inserts** option, see *Change Processing Tuning (page 1181)*.

No record found for applying an UPDATE: Click the triangle to open the list and select what happens when there is a conflict with an UPDATE operation. You can select one of the following from the list:

- **Ignore record:** The task continues and the error is ignored.
- **Log record to the exceptions table** (default): The task continues and the record is written to the exceptions table.
- **Suspend table:** The task continues but data from the table with the error record is moved into an error state and its data is not replicated
- **Stop task:** The task is stopped and manual intervention is required.
- **Insert the missing target record:** The missing target record will be inserted into the target table. When the source endpoint is Oracle, selecting this option requires supplemental logging to be enabled for all the source table columns.

Escalate handling when apply conflicts reach (per table): Select this check box to escalate error handling when the number of apply conflicts (per table) reaches the specified amount. Valid values are 1-10,000.



- *When this option is selected and the number of conflicts reaches the specified number,*
- *Apply Conflicts errors are counted for each table separately when a task is run. When a task stops, the error count reverts to zero. When a task is resumed/reloaded, the number of errors for all the tables are reset.*
- *When working in [Batch optimized apply](#) Change Processing mode, the calculation of the Apply Conflicts amount does not include DELETE and UPDATE conflicts that were ignored (as a result of enabling the **Ignore Record** option described above).*

Escalation action: Choose what action Replicate should perform when handling is escalated. Note that the available actions are dependent on the action selected in the [drop-down lists described above](#).



When this option is selected and the number of conflicts reaches the specified number, only the escalation action will be performed (i.e. the original action set for the Apply conflict will not be performed).

- **Log record to the exceptions table** (default): The task continues and the error is written to the task log and to the exceptions table.
- **Suspend table:** The task continues but data from the table with the error record is moved into an error state and its data is not replicated.
- **Stop task:** The task is stopped and manual intervention is required.

Logging

You can set the logging level for task logs by selecting the **Logging** tab in the **Task Settings** dialog box and then selecting the **Logging Level** sub-tab. The level you set determines what information is written to the log.



You can also set the task logging level from the **Tools** menu in Monitor view.

For more information, see *Monitor mode (page 95)* and *Setting the task logging level (page 1234)*.

For more information on monitoring, see *Monitor mode (page 95)*. For more information on setting the task logging level in Qlik Replicate, see the *Qlik Replicate Setup and User Guide*.

The following are the available logging levels. The list is in order from the lowest level to the highest level.

1. Error
2. Warning
3. Info
4. Trace
5. Verbose

The higher levels always include the messages from the lower levels. Therefore, if you select **Error**, only error messages are written to the log. However, if you select **Info**, informational messages, warnings, and error messages are included. Selecting **Verbose** writes all possible messages to the log.

For information on how to set the logging level, see *Setting the task logging level (page 1234)*.

Storing trace and verbose logging in memory

When the logging level is set to "Trace" or "Verbose", you can instruct Replicate to store the logging information in memory until an error occurs. On detecting an error, Replicate will begin writing to the physical logs and continue to do so for a few minutes after the initial occurrence of the error.

If *no* error occurs before the allocated memory is used up, Replicate will empty the memory buffer and start afresh.

This option is useful for tasks that fail unpredictably and for no obvious reason. The problem with continually writing large amounts of information to the logs is twofold:

- Running in "Trace" or "Verbose" logging mode will quickly use up available disk space (unless the logging settings have been configured to prevent this).
- Continually writing large amounts of data to the logs will affect performance.

To use this option

1. Select the **Store trace/verbose logging in memory, but if an error occurs, write to the logs** check box at the top of the tab.
2. In the **Allocate memory up to (MB)** field, specify the amount of memory you want to allocate for storing logging information.

Character substitution

You can substitute or delete source characters in the target database and/or you can substitute or delete source characters that are not supported by a selected character set.



- All characters must be specified as Unicode code points.
- Character substitution will also be performed on Replicate Control Tables.
- Invalid values will be indicated by a red triangle in the top right of the table cell. Hovering your mouse cursor over the triangle will show the error message.
- Any table-level or global transformations defined for the task will be performed after the character substitution has been completed.
- Substitutions actions defined in the **Substitute or Delete Source Characters** table are performed before the substitution action defined in the **Substitute or Delete Source Characters Unsupported by the Selected Character Set** table.
- Character substitution does not support LOB data types.

Substituting or deleting Source Characters

Use the **Substitute or Delete Source Characters** table to define replacements for specific source characters. This may be useful, for example, when the Unicode representation of a character is different on the source and target platforms. For example, on Linux, the minus character in the `SHIFT_JIS` character set is represented as `U+2212`, but on Windows it is represented as `U+FF0D`.


Substitution actions

To	Do This
Define substitution actions.	<ol style="list-style-type: none"> 1. Click the Add Character button above the table. 2. Specify a source character and a target character in the Source Character and Substitute Character fields respectively. For example to replace the letter "a" with the letter "e", specify 0061 and 0065 respectively. <div data-bbox="635 1462 1390 1597" style="border: 1px solid #ccc; padding: 5px; margin: 10px 0;"> To delete the specified source character, enter 0 in the Substitute Character column. </div> 3. Repeat steps 1-2 to replace or delete additional characters.
Edit the specified source or target character	Click anywhere in the relevant column and change the character as required.
Delete entries from the table	Select the desired entry or entries and click the Delete button.

Substituting or deleting Source Characters unsupported by the selected character set

Use the **Substitute or Delete Source Characters Unsupported by the Selected Character Set** table to define a single replacement character for all characters not supported by the selected character set.

Unsupported character substitution actions

To	Do This
Define or edit a substitution action.	<ol style="list-style-type: none"> 1. Select a character set from the Character Set drop-down list in the table. Any characters not supported by the selected character set will be replaced on the target by the character specified in step 2 below. 2. In the Substitute Character column, click anywhere in the column and specify the replacement character. For example, to replace all unsupported characters with the letter "a", enter 0061. <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">  <i>To delete all unsupported characters, enter 0.</i> </div>
Disable the substitution action.	Select the blank entry from the Character Set drop-down list.

File uploads

Note that the **File Uploads** tab will only be shown if the task is defined with an endpoint that supports this feature.

Click the **Optimize File Uploads** button to improve performance when replicating to file-based targets such as Amazon S3 and Hadoop. When this feature is enabled, the button text changes to **Disable File Upload Optimization**. Click the **Disable File Upload Optimization** button to disable file upload optimization.

The upload mode depends on the task type:

- **Full Load** - Multiple files created from the same table are transferred in parallel, in no particular order.
- **Apply Changes** - Files created from multiple tables are transferred in parallel. Files created from the same table are transferred sequentially according to creation time.
- **Change Data Partitioning** - Files created from multiple tables and files created from the same table are transferred in parallel.

Note that disabling this option after the task has already started will require you to do one of the following:

- If the task is in the Full Load stage, reload the target using the **Reload Target Run** option.
- If the task is in the Change Processing stage, resume the task using the **Start processing changes from Run** option.



- Supported by the following target endpoints only: Amazon S3, Hadoop (Hortonworks, Cloudera, and MapR) Microsoft Azure ADLS, Databricks (Cloud Storage), Microsoft Azure HDInsight, Hortonworks Data Platform (HDP), Google Cloud Storage, Google Cloud Dataproc, Amazon EMR, and Cloudera Data Platform (CDP) Private Cloud.
- **General Limitations and Considerations:**
 - **Post Upload Processing** endpoint settings are not supported.
- **Hadoop - Limitations and Considerations:**
 - When replicating to a Hadoop target, only **Text** and **Sequence** file formats are supported.
 - Hive jobs are not supported as they will prevent the file upload.
 - Append is not supported when using **Text** file format.
- **Amazon S3 and Microsoft Azure ADLS - Limitations and Considerations:**
 - When working with Reference Files, a new entry is added to the Reference File immediately after the data file is uploaded (even if the DFM file has not been uploaded yet).
 - The existence of the DFM file does not necessarily mean that the associated data file has also been uploaded.

Message Format

The **Message Format** tab will only be shown if the task is defined with a streaming target endpoint that supports this feature.

When a task is defined with such an endpoint, you can specify a custom message format that will override the default Replicate message format. This may be useful if the consumer application needs to process the message in a particular format.

The custom message format can be defined at task level and/or at table level. When it is defined at both task *and* table level, the message format defined for the table will take precedence over the message format defined for the task.

For information on defining a custom message format at table level, see *Message format (page 1106)*.



- Supported by the Kafka target endpoint only
- The custom message format feature cannot be used in conjunction the Attunity Avro Message Decoder SDK
- Hierarchical structured messages are not supported.

General rules and usage guidelines

When defining a custom message, it is import to consider the rules and usage guidelines listed below.

Section names:

The following naming rules apply:

- Section names must start with the characters a-z, A-Z, or _ (an underscore) and can then be followed by any of the following characters: a-z, A-Z, 0-9, _
- With the exception of the **Record name** and **Key name sections** (that do not end with a slash), removing the slash from section names will flatten the hierarchy of the associated section (see [Slashes](#) below).
- All section names except **Record name** and **Key name** can be deleted (see *Deletion: (page 1198)* below)
- The **Data name** and **Include record Before-data** section names cannot both be deleted
- The **Data name** and **Include record Before-data** section names cannot be the same

Slashes:

Some of the section names in the UI end with a slash (e.g. beforeData/). The purpose of the slash is to maintain a hierarchy of the different sections within the message. If the slash is removed, the following will occur:

- The hierarchical structure of that section will be flattened, resulting in the section name being removed from the message
- The section name will be prefixed to the actual metadata, either directly or using a separator character (e.g. an underscore) that you appended to the name

Example of a data message when headers/ is specified with a slash:

```
"message":{
  "data":{
    "COL1": "159",
    "COL2": "159"
  },
  "beforeData": null,
  "headers": {
    "operation": "INSERT",
    "changeSequence": "2018100811484900000000233",
```

Example of a data message when headers_ is specified with an underscore instead of a slash:

```
"message":{
  "data":{
    "COL1": "159",
    "COL2": "159"
  },
  "beforeData": null,
  "headers_operation": "INSERT",
  "headers_changeSequence": "2018100811484900000000233",
```

Deletion:

Deleting a section name from the message will flatten the hierarchical structure of that section. This will result in all of that section's data appearing immediately below the content of the preceding section.


Example of a data message with the `headers` section name:

```
"message":{
  "data":{
    "COL1": "159",
    "COL2": "159"
  },
  "headers": {
    "operation": "INSERT",
    "changeSequence": "2018100811484900000000233",
```

Example of a data message without the `headers` section name:

```
"message":{
  "data":{
    "COL1": "159",
    "COL2": "159"
  },
  "beforeData": null,
  "operation": "INSERT",
  "changeSequence": "2018100811484900000000233",
```

Variables

You can add variables to names by clicking the  button at the end of the row. The following variables are available:

- SERVER_NAME - The host name of the Replicate Server
- TARGET_TABLE_NAME - The name of the table
- TARGET_TABLE_OWNER - The table owner
- TASK_NAME - The name of the Replicate task




The `TARGET_TABLE_OWNER` variable is not available for the **Record name** and **Key name** options (described in the table below).

Defining a custom message format

To define a custom message format, click the **Custom Message Format** button and configure the options as described in the table below.

To revert to the default message format, click the **Default Message Format** button.

Message format options

Option	Description
Include Namespace	<p>A unique identifier for the message. This should be a string, separated by periods. Note that the Namespace will be included in both the message and the message key.</p> <p>Example:</p> <p>mycompany.queue.msg</p> <p>Default name:</p> <p>com.attunity.queue.msg.{{TASK_NAME}}.{{TARGET_TABLE_OWNER}}.{{TARGET_TABLE_NAME}}</p> <p>Default name for Control Tables:</p> <p>com.attunity.queue.msg.{{TARGET_TABLE_NAME}}</p> <div data-bbox="347 795 1390 1720" style="border: 1px solid #ccc; padding: 10px; margin-top: 10px;"> <p> The default Control Table Namespace cannot be changed via the UI. You can, however, change the default Control Table Namespace as follows:</p> <ol style="list-style-type: none"> 1. In the Task Settings Message Format tab, click the Custom Message Format button and then click OK. 2. Save the task and then export it using the Export Task toolbar button. 3. Open the exported JSON file and add the <code>control_table_namespace</code> parameter as follows (replacing <code>MyNamespace</code> with your own value): <pre style="font-family: monospace; font-size: 0.9em;"> "task_settings": { "source_settings": { }, "target_settings": { "queue_settings": { "use_custom_message": true, "message_shape": { "control_table_namespace": "MyNameSpace" }, "use_custom_key": true, "key_shape": { </pre> <ol style="list-style-type: none"> 4. Save the JSON file and then import it to Replicate using the Import Task toolbar button. <p style="text-align: center; font-style: italic;">For instructions on exporting/importing tasks, see Migrating tasks (page 126).</p> </div>
Record name	<p>The name of the record (i.e. message).</p> <p>Default name:</p> <p>DataRecord</p>

Option	Description
Data name	All of the data columns included in the record. Default name: data/
Include Headers	Header columns provide additional information about the source operations. For more information, see <i>Include the following headers (page 1200)</i> below.
Include headers namespace	A unique identifier for the header columns section. This should be a string, separated by periods. Example: headers.queue.msg Default name: com.attunity.queue.msg
Headers name	The name of the section containing the Replicate column headers. Default name: headers/
Include the following headers	For a description of the available header columns, see Data Messages in <i>Metadata and data messages (page 838)</i> . Default: All columns are included by default, except the External Schema ID column.
Include record Before-data	When this check box is selected (the default), both pre and post UPDATE data will be included in UPDATE messages. To include only the post UPDATE data in messages, clear the check box. Default name: beforeData/

Option	Description
Key name	<p>The name of the section containing the message key.</p> <p>Default name:</p> <p>keyRecord</p> <p>This option is only applicable when:</p> <ul style="list-style-type: none"> • The Encode the message key in Avro format option is enabled • The Metadata Publishing Method is set to Publish data schemas to Confluent Registry Schema or Publish data schemas to Hortonworks Schema Registry • The Message key is not set to None

Transformations and Filters

Click the **Transformations and Filters** tab to show the **Expression defaults** tab. Expressions can be used in global transformations and filter rules as well as in table-level transformations and filters.

This option can be set globally for all tasks or individually for a specific task. To set the option for a specific task, click the **Change to Task Policy** button and set the option as required. If you want to use the global policy set in the server settings' *Transformations and Filters (page 1281)* tab, click the **Change to Global Policy** button.

- **Support special characters in column names used in expressions:** Enable this option if you want to include source column names with special characters in expressions defined for this task. An example of such a column name would be `special#column`.



A condition for enabling this option is that all table and global transformations must use strict SQL syntax. Specifically, all string literals must use single quotes (') rather than double quotes. If any of the transformations on any task on the server uses double quotes (") to quote literals, then you should not enable this option.

Before enabling this option, you must fix any use of double quotes in string literals. For example, a transformation such as:

```
CASE age WHEN age < 18 THEN "adult" ELSE "minor" END)
```

must be rewritten as:

```
CASE age WHEN age < 18 THEN 'adult' ELSE 'minor' END)
```

See also:

- *Defining global rules (page 1107)*
- *Using filters (page 1092)*
- *Defining transformations for a single table/view (page 1084)*

- *Using the Expression Builder (page 1137)*

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

14 Working with tasks at runtime

This section describes how to work with tasks that you design. For information on how to design a task, see *Defining and managing tasks (page 109)*. This chapter contains information on running tasks, viewing the task status, and viewing messages about the task. Information on monitoring and working with tasks during runtime is in the section *Monitoring and controlling replication tasks (page 1216)*.

In this section:

- *Running a task (page 1203)*
- *Viewing the task status (page 1212)*
- *Accessing task messages and notifications (page 1212)*

14.1 Running a task

After you design a task (see *Defining and managing tasks (page 109)*), you can run and monitor its progress with one click in Qlik Replicate. This simple Click-2-Replicate function is described in this section. In addition, the various types of run options available are also described.

In this section:

- *How to run a task (page 1203)*
- *Using the Run button options (page 1204)*
- *Advanced Run Options (page 1205)*
- *Recovering from data folder loss or corruption (page 1210)*



The task run buttons area available in the toolbar at the top of the console in the following views:

- *Tasks view (page 91) (in both Designer mode (page 93) and Monitor mode (page 95))*
- *When Viewing specific tasks (page 93)*

How to run a task

Click the **Run** button to execute a replication task. The task process continues to run until you click the **Stop** button to stop the task.



When you click **Run**, the following occurs:

- If this is the first time that a task is run, the **Start Processing** operation is run.
- If the task has been started and stopped, the **Resume Processing** operation described in *Using the Run button options* (page 1204) is run.
- If changes were made to the endpoint, change processing takes place after the full load operation. If you do not want change processing to occur or if you want to start change processing from a predetermined point, you must make the appropriate *Advanced Run Options* (page 1205) selection.

In some cases, task replication may stop due to an error although the task process is still running.

See *Tasks view* (page 91) for information on the task status and how Qlik Replicate displays information on the current task status.

The **Run** button is available in the following views:

- The Tasks view when you select a task from the Task List.
- For the individual task, both the Designer mode and Monitor mode have the **Run** and **Stop** buttons available.



You must be in the Monitor mode to view the task progress.

Using the Run button options

Clicking the **Run** button runs a full-load replication task from the source to the target. This is a first time task that creates the target endpoints and loads the source data to the target according to your task definitions.

Subsequent runs allow you to resume processing from a specific point and process changes. In addition, you can also specify from what point you want the replication to start.

The following options are available:

- **Start Processing** - This is available the first time you run the task only. This will execute the initial full load operation. If **Change Processing** is also enabled for the task or if it is an Apply Changes only task type, change processing will start as soon as any changes are made to the source endpoint.(switches to **Resume Processing** after the task has started)
- **Resume Processing** - Resumes task execution from the point that it was stopped. You can also resume processing by clicking the **Run** button if the task has been stopped.



If the schema or a filter was changed after the task stopped, the task should be reloaded as opposed to resumed (see below).

- **Reload Target** (Only available when the **Full Load** or **Full Load** and **Apply Changes** replication options are enabled) - Starts the Full Load and Change Processing (if enabled) from the beginning. Tables that have already been processed are handled according to the [relevant "Target table preparation" setting](#).



To replicate tables that were added to the local file channel task after the initial full load, you need to reload both the local and the remote file channel tasks.

- **Advanced run options** - Advanced options for controlling task execution. For more information, see *Advanced Run Options (page 1205)*

Advanced Run Options

Advanced Run Options provide you with additional options for resuming and restarting tasks.



Some of the advanced run options are not available in a Log Stream Staging setup.

For information on the availability of advanced run options in a Log Stream Staging setup, see [Log Stream Staging Limitations and Considerations](#).

To use Advanced Run Options, click the triangle next to the **Run** button and select **Advanced Run Options**.

The **Advanced Run Options** dialog box opens.

The **Advanced Run Options** dialog box lets you do the following:

- ****Restart task and start processing changes from current time:** This starts the Apply Changes replication task from the beginning (as if the task has not run before).
****Only available for Apply Changes replication tasks.**
- **Tables are already loaded. Start processing changes from:**



Metadata changes performed on the source tables while a task is stopped (for example, DROP COLUMN) will not be captured when the task is resumed from an earlier time or position (SCN/LSN). In such a case, the metadata that exists when the task is resumed will be captured.

- **Date and Time:** Select the date and time from where you want to Replicate to start processing changes.



- When logs are deleted from the database (e.g. due to a purge policy), a log matching the specified date and time may not exist. In this case, Replicate will resume the task from the earliest point possible, after the specified date and time.
- With the IBM DB2 for LUW source endpoint, this option cannot be used to start Apply Changes tasks from a point before the original start time of the Apply Changes task. Or, to put it another way, it can only be used to start tasks from any time after the original start time of the Apply Changes task.
- The timestamp uses the local time of the browser machine.
- This option is not relevant for the File Source endpoint.

- **Source change position (e.g. SCN or LSN):** Specify the position in the log from where to resume change processing. The source change position format differs according to your source endpoint. For more information, see *How to Find the Source Change Position* (page 1209).



The Source change position option is supported with the following source endpoints only:

- Oracle
- Microsoft SQL Server
- MySQL
- PostgreSQL
- IBM DB2 for z/OS
- IBM Informix
- IBM DB2 for LUW

Metadata Only:

The "Metadata only" options described below allow you to:

- Create empty tables on the target and then manually edit them.
- Create tables during a task.

Enabling the options will also ensure that supplemental logging is set up correctly on the source tables before starting the actual replication task.



The "Metadata only" feature is not supported when the task is defined with the Apply Changes only [task option](#). any of the following [task options](#):

- Apply Changes only
- Store Changes only
- Apply Changes and Store Changes

- **Recreate all tables and stop:** Select this option to recreate the target tables as defined in the **Full Load Settings** tab. When "Store Changes" is enabled, the Change tables/Audit table will be created as defined in the **Store Changes Settings** tab. To use this option, stop the existing task, run the task with this option enabled (the task will stop automatically) and finally, resume the task.
- **Create missing tables and stop:** Select this option to create missing target tables including Change Tables. You can use this option to create Change Tables on the target after enabling the "Store Changes" option (in the **Store Changes Settings** tab) for an existing task. To use this option, stop the existing task, run the task with this option enabled (the task will stop automatically) and finally, resume the task.

The table below shows which tables are created in the target database when the **Metadata only** option is enabled for a unidirectional task. As the table shows, when certain task options are enabled, [Control tables](#) and the [Audit table](#) will not be created on the target.

Enabled Task Options	Tables Created on the Target
Full Load	All tables except for Control tables
Full Load and Apply Changes	All tables except for Control tables
Full Load, Apply Changes and Store Changes - When changes are stored in Change tables	All tables except for Control tables
Full Load and Store Changes - When changes are stored in an Audit table	All tables except for Control tables and the Audit table
Full Load, Apply Changes and Store Changes - When changes are stored in an Audit table	All tables except for Control tables and the Audit table

Recovery:

- **Recover using locally stored checkpoint:** Use this option if recovery is not possible using the **Resume Processing** or **Start process changes from** options (due to corrupt swap files, for example). When this option is selected, Replicate uses the checkpoint data stored in <Data_Folder_Path>\data\tasks\



When using this option, the following limitations apply:

- The following source endpoints are supported only:
 - Oracle
 - Microsoft SQL Server
 - MySQL
 - PostgreSQL
 - IBM DB2 for z/OS
 - SAP HANA
- Tasks can only be recovered during Change Processing (i.e. after Full Load Completes)
- With the exception of the File Channel endpoint, all target endpoints are supported. The following limitations apply:
 - **In [Transactional apply](#) Change Processing mode:** All target endpoints that support transactions are supported.
 - **In [Batch optimized apply](#) Change Processing mode:** Oracle target endpoint only is supported. Also requires the **Preserve transactional integrity** option to be enabled.
 - For all other target endpoints or Change Processing modes, recovery is supported, but may cause duplicates on the target.

- **Recover using checkpoint stored on target:** Select to recover a task using the CHECKPOINT value from the attrep_txn_state table (created in the target database).



When using this option, the following limitations apply:

- Only the following source and target endpoints are supported:
 - Oracle
 - Microsoft SQL Server
- Tasks can only be recovered during Change Processing (i.e. after Full Load Completes)
- The task Change Processing mode must be set to either:
 - [Batch optimized apply](#) with the **Preserve transactional integrity** option enabled. Note that this mode is only supported with the Oracle target endpoint.
 - OR
 - [Transactional apply](#)
 For information about setting the Change Processing mode, see [Changes Processing Tuning](#).

This option will only be available if the **Store task recovery data in target database** option was enabled in the Task Settings' *Change Processing Tuning* (page 1181) tab before Change Processing completed.

Select this option (as opposed to the **Recover using locally stored checkpoint** option) if the files in the **Data** folder are corrupted or if the storage device containing the **Data** folder has failed.

For a detailed explanation of how to set up and implement recovery using the `attrep_txn_state` table, see *Recovering from data folder loss or corruption (page 1210)*.

How to Find the Source Change Position

The following topic explains how to locate the source change position for each of the supported databases.

MySQL

OPTION 1:

1. Execute:
`SHOW BINARY LOGS;`
2. Choose a binlog file (e.g. `log.000123`).
3. Execute:
`SHOW BINLOG EVENTS IN 'binlog_filename';`
Example:
`SHOW BINLOG EVENTS IN 'log.000123';`
4. Locate the position of a BEGIN or DDL statement (e.g. `1777888`).



MySQL does not support a "start from position" located in the middle of a transaction, which is why you must select either BEGIN or DDL.

5. Set the internal parameter:
`StartFromPosition = binlog_filename:begin_or_ddl_position`
Example:
`StartFromPosition = log.000123:1777888`

OPTION 2:

1. Execute:
`SHOW MASTER STATUS;`
2. Set the "Start from position" as:
`firstColumnValue:secondColumnValue`
Example:
`mysql.007472:775`

Microsoft SQL Server

Execute:

```
SELECT MAX([Current LSN]) FROM fn_dblog(null,null);
```

Example: 00002717:00000e08:0003

Oracle

Execute:

```
SELECT current_scn FROM V$DATABASE;
```

Example: 1471212002

PostgreSQL

Note that the command differs according to your PostgreSQL version.

PostgreSQL 9.6

Execute:

```
SELECT * FROM pg_current_xlog_location();
```

PostgreSQL 10 and later

Execute:

```
SELECT * FROM pg_current_wal_lsn();
```

IBM DB2 for z/OS

Specify the LSN, which is the RBA in a non data sharing environment, and the LRSN in a data sharing environment

IBM DB2 for LUW

1. Execute db2pd to find the correct log name.
2. Execute db2f1sn with the returned log name and use the current LRI value.
See also: *Resuming or starting a task from LRI in a pureScale environment (page 274)*

Recovering from data folder loss or corruption

During normal operation, Qlik Replicate maintains the replication state in the following location:

```
<Data_Folder_Path>\data\tasks\<task_name>\StateManager
```

This enables tasks that cannot be resumed normally (due to corrupt swap files, for example) to be recovered using the **Recover using locally stored checkpoint** option described in *Advanced Run Options (page 1205)*.

However, if the files in the **data** folder become corrupted or if the storage device containing the **data** folder fails, tasks must be recovered using the means described below.

This option is supported when replicating between the following endpoints only:

Supported source and target endpoints

Supported source endpoints	Supported target endpoints
----------------------------	----------------------------

<ul style="list-style-type: none">• Amazon RDS for MySQL• Amazon RDS for PostgreSQL• Amazon RDS for SQL• AWS Aurora Cloud for PostgreSQL• Google Cloud SQL for MySQL• Google Cloud SQL for PostgreSQL• IBM DB2 for z/OS• Microsoft Azure Database for MySQL• Microsoft Azure SQL Managed Instance• Microsoft SQL Server• MySQL• Oracle• PostgreSQL• SAP HANA	<ul style="list-style-type: none">• Amazon Redshift• Google Cloud SQL for PostgreSQL• Microsoft Azure Database for MySQL• Microsoft Azure Database for PostgreSQL• Microsoft SQL Server• MySQL• Oracle• PostgreSQL• Snowflake on Azure• Snowflake on Google• Snowflake on AWS
---	---

Setting up and initiating task recovery

For recovery to be successful, the source database transaction logs must be available from the time the task failed.

To set up a task for recovery

1. Design a task. Make sure to enable the **Store task recovery data in target database** option in the Task Settings' *Change Processing Tuning* (page 1181) tab. This option can be enabled at any time during Change Processing, although it must be enabled before Change Processing completes.
2. Export the task definitions as described *Exporting tasks* (page 126).
3. Run the task.

In addition to the selected source tables, the task will write the checkpoint data to the following table in the target database (and automatically create the table if it has not already been created by another task):

attrep_txn_state

To initiate recovery

1. Import the task definition exported when you set up the task.
2. Enter the passwords in the endpoint connection settings.
3. Access the attrep_txn_state table on the target database and locate the failed task in the TASK_NAME column. If there are tasks with the same name running on multiple Replicate Servers, you will also need to locate the appropriate server in the SERVER_NAME column. After locating the relevant task, copy the value in the corresponding CHECKPOINT column.
4. Select the **Recover using checkpoint stored on target** option and then provide the CHECKPOINT value (preferably by pasting) as described in *Advanced Run Options* (page 1205).
5. Click **OK** to start the recovery.

During recovery, Replicate does not write anything to the target database until it identifies the commit event corresponding to the CHECKPOINT value. Once it identifies the CHECKPOINT commit event, recovery is performed and the task reverts to standard operation.



14.2 Viewing the task status

In the Tasks View, you can see the task status by viewing the icon for the task. After a task is run, the task icons in the Task view display the current status of the task. For additional information on the possible statuses, see *Tasks view (page 91)*.

The following icon represents a task that is in an error status.

There are two types of errors:

Error icon descriptions

Error icon	Description
	<p>Recoverable error: A recoverable error indicates that there is a temporary problem, such as an inability to connect to one of the endpoints. The task icon is blue indicating that the task is still active. In this case, Qlik Replicate attempts to restart the task according to the <i>Error handling (page 1186)</i> settings.</p> <p>The task remains active but paused throughout the error state. You can stop the task at any time and resolve the error manually, if necessary.</p> <p>See also <i>Error handling (page 1186)</i>.</p>
	<p>Fatal Error: When a fatal error occurs, the task stops and you must resolve the error manually. You cannot start the task again until the error is resolved. Use the logs or the messages in the Alerts pane to see the error type.</p> <p>See also:</p> <p><i>View Log Messages for a task (page 1214)</i></p> <p><i>Viewing Notifications (page 1213)</i></p>

14.3 Accessing task messages and notifications

Task messages are displayed in the **Messages** section of the Qlik Replicate Console. The **Messages** section is located at the bottom right of the console in the *Monitor mode (page 95)* and when *Viewing specific tasks (page 93)*.

The Message section has two types of messages that provide information about events that occur in a task. Each type of message is displayed in the following tabs:

- *Viewing Notifications (page 1213)*
- *View Log Messages for a task (page 1214)*

Viewing Notifications

The **Notifications** tab displays notifications about the task. These messages alert you to specific events encountered by a task, such as the task starting or stopping, a specific error type, or information about latency and disk space.

The **Notifications** tab displays the time of a notification and a description of the notification. You define the notifications that are sent for each task and a description for each notification in the Settings area. For more information, see *Creating a new notification* (page 1240).

Using the Notifications list

When a notification is sent, it is displayed in the **Notifications** tab. This section describes the tasks that can be performed in the **Notifications** tab.

Opening a Notification

When you open a notification, you can see the full message presented in a dialog box. The dialog box contains a button to copy the text so that you can use it somewhere else for troubleshooting and the timestamp for the notification.

To open a notification:

1. In the Messages section of the console, click the **Notifications** tab. The **Notifications** tab opens.
2. Select the notification you want to open from the list.
3. Double-click the notification or click **Open** from the toolbar at the top of the list.

Clearing a Notification

You can clear notifications from the list to make sure that you are seeing only those that are relevant to you.

To clear a notification:

1. In the Messages section of the console, click the **Notifications** tab.
2. Select the notification you want to clear from the list.
3. Click **Clear** from the toolbar at the top of the list.

Sorting Notifications

You can sort log messages according to Date and Time and Message.

To sort the notifications:

1. In the **Messages** section of the console, click the **Notifications** tab.
2. Click the **Date and Time** or **Message** column according to how you want to sort the messages. An upward arrow indicates that the column is sorted in ascending order whereas a downward arrow indicates that the column is sorted in descending order.

View Log Messages for a task

The **Log Messages** tab displays log messages for errors or warnings from a task. The errors are listed in this tab with the time of the error or warning and the log entry for the event. You can choose to view both errors and warnings or only one of them.

If errors or warnings exist in the task, a red circle with the total number of errors and warnings is displayed. The number displayed may be the number of errors, the number of warnings, or the total of number of errors and warnings depending on what you select to view in this tab. The **Log Messages** tab is shown in the figure below.

Using the Log Messages list

When a log error or warning is sent, it is displayed in the **Log Messages** tab. This section describes the tasks that can be performed in the **Log Messages** tab.

Selecting the log message type

Two types of log messages are displayed in the Log Messages list. You can view Errors, Warnings, or both.

To select the log message type:

- Select the check box or boxes for the type messages you want to view. The check boxes are located at the top of the Log Messages List.

Opening a log message

When you open a log message, you can see the full log text presented in a dialog box. The dialog box contains a button to copy the text so that you can use it somewhere else for trouble shooting and the timestamp for the log message.

To open a log message:

1. In the **Messages** section of the console, click the **Log Messages** tab.
2. Select the log message you want to open from the list.
3. Double-click the log message or click **Open** from the toolbar at the top of the list.

Clearing a log message

You can clear log messages from the list to make sure that you are seeing only those that are relevant to you.

To clear a log message:

1. In the Messages section of the console, click the **Log Messages** tab.
2. Select the log message you want to clear from the list.
3. Click **Clear** from the toolbar at the top of the list.

Sorting log messages

You can sort log messages according to Date and Time, Level and Message.

To sort the log messages:

1. In the Messages section of the console, click the **Log Messages** tab.
2. Click the **Date and Time**, **Level** or **Message** column according to how you want to sort the messages. An upward arrow indicates that the column is sorted in ascending order whereas a downward arrow indicates that the column is sorted in descending order.

Viewing the log file in the Log Viewer

In addition to viewing the log messages, you can view the entire log file in the log viewer.

To view the log in the Log Viewer:

- From the **Messages** section, click **View Logs**.
The Log Viewer opens.

For a description of actions you can perform in the Log Viewer, see *Viewing, downloading, and manually rolling over task log files* (page 1235).

15 Monitoring and controlling replication tasks

When you monitor and run a task, you can use the Click-2-Replicate function to carry out the replication task and view its functions in near real time. This section describes how to run and monitor a replication task.

In this section:

- *Viewing information in the monitor (page 1216)*
- *Monitoring full-load operations (page 1216)*
- *Monitoring change processing operations (page 1224)*
- *Viewing messages (page 1233)*
- *Using the monitor tools (page 1233)*

15.1 Viewing information in the monitor

You access the Monitor view when you open a specific task. The monitor provides near real-time information for the task you select.

To access the Monitor:

1. When *Viewing specific tasks (page 93)*, select the task you want to monitor.
2. From the toolbar at the top of the console, click **Open**.
3. From the toolbar at the top right, click **Monitor**.
The Monitor opens. To view the information in real time, you need to run the task (if the task has not already started). For information on running a task, see *Running a task (page 1203)*.

15.2 Monitoring full-load operations

You can view the progress of a full-load operation in the left side of the Monitor.

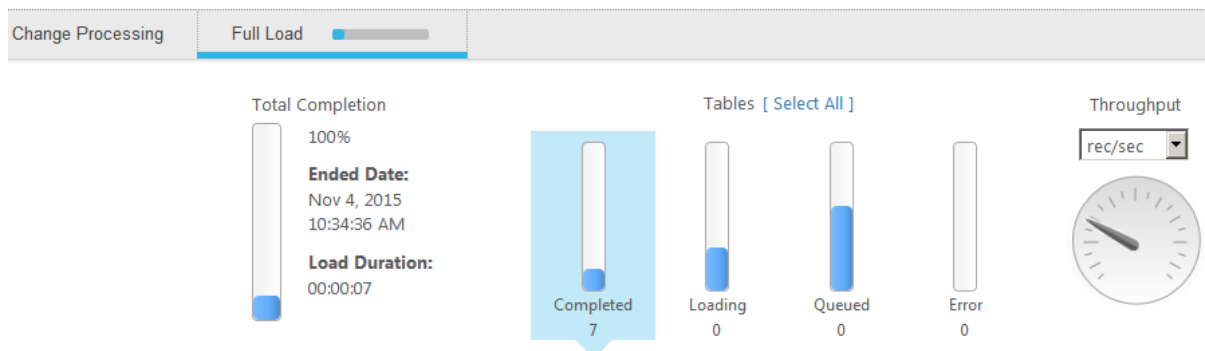
To make sure you are viewing the information for a full-load operation, select the **Full Load** tab.

You can view the following:

- *General information for a Full Load (page 1216)*
- *Detailed information for the Full Load (page 1217)*
- *Monitoring throughput in a Full Load operation (page 1224)*

General information for a Full Load

General information about the Full Load is presented in a graphical format. The following figure shows the graphical information displayed when a task is running.



This section has the following information:

- **Status bars:** Indicates the status of the tables being loaded.
 - **Completed:** The number of tables that finished loading into the target endpoint.
 - **Loading:** The number of tables that are in the process of loading into the target endpoint.
 - **Queued:** The number of tables that are waiting to load into the target endpoint.
 - **Error:** The number of tables that could not be loaded due to an error. See *Accessing task messages and notifications (page 1212)* for information about error messages.
- **Full-load total completion bar:** Displays the progress of all records being loaded to the target endpoint. The bar is located in the **Full Load** tab at the top of the graph section.
- **Throughput gauge:** Displays the current throughput. Throughput displays the number of events read in the task for a specified amount of time.

You can also view *Detailed information for the Full Load (page 1217)*.

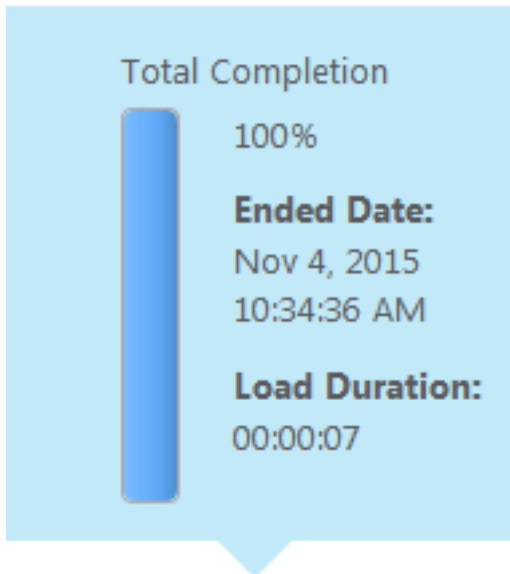
Detailed information for the Full Load

For each of the status bars displayed in the *General information for a Full Load (page 1216)* graphs, a table is displayed in the section below with specific information about the current loading status. The following information is available:

- *General information for a completed task (page 1217)*
- *Information for each table in the task (page 1218)*
- *Information for tables that have completed loading (page 1220)*
- *Information for tables that are currently loading (page 1221)*
- *Information for tables that are in the loading queue (page 1223)*
- *Information for tables with errors (page 1223)*

General information for a completed task

This section displays a table with information for all of the completed tables in a task. To view this table, click the **Total Completion** bar, shown in the figure below.



Total Completion Status

This table displays the following Progress Details:

Progress details

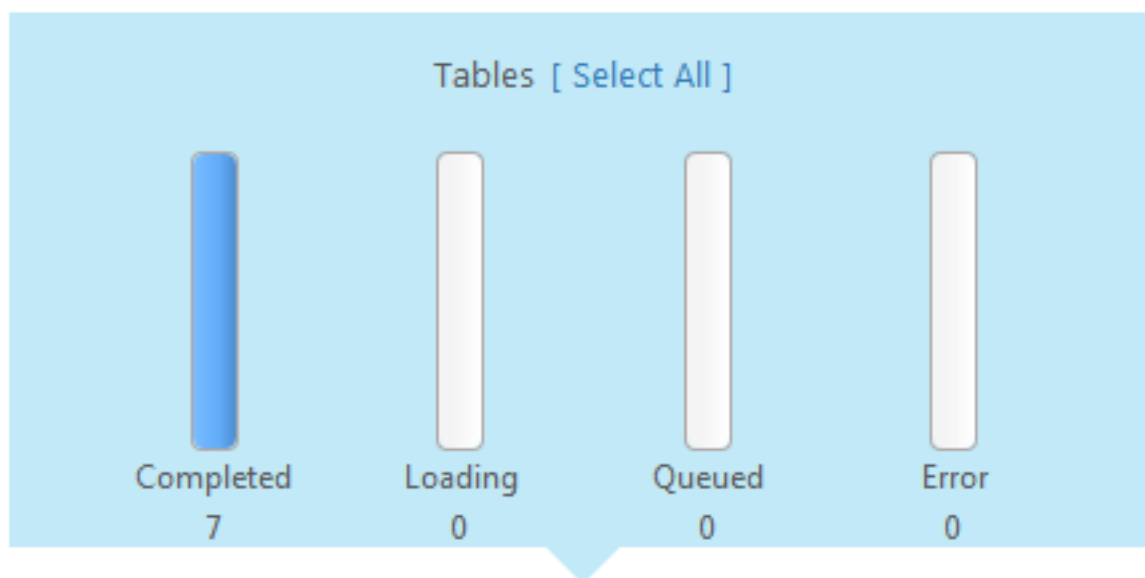
Detail	Total	Completed	Remaining	Notes
Tables	The total number of tables that are included in the task.	The total number of tables that completed loading at the current time.	The total number of tables waiting to be loaded.	Additional information.
Records	The total records that completed loading at the current time.	The total number of records that completed loading at the current time.	The total number of records waiting to be loaded.	Additional information.
Time	The estimated time to load all of the selected tables in the task.	The total elapsed time.	The estimated amount of time to load the remaining tables.	Additional information.



Adding a table and then reloading the task or simply reloading the task will not affect the Full Load start/end time or duration as these times always pertain to the original Full Load.

Information for each table in the task

This section describes the progress of each of the tables being processed for the task. To display this information, click the **[Select all]** link above the



Select All Tables

The information is displayed in a table that has the following columns:

- **Table Name:** The names of the source tables that are included in the task.
- **Status:** This is a statement that describes the status for the table. The following are the statuses that can be displayed:
 - **Queued:** The table is in the queue waiting to be loaded to the target endpoint.
 - **Loading:** The table is being processed but is not finished loading.
 - **Completed:** All of the table records are loaded to the target.
 - **Error:** The table stopped loading due to an error. See *Accessing task messages and notifications* (page 1212) for more information about task errors.
- **Estimated Count:** The number of records that are loaded to the target.
- **Elapsed Time:** The total elapsed time since the table records began processing.
- **Progress:** The table status and the time the table entered that status.

Reloading or unsuspending tables

You can reload or unsuspend a table, depending on the current task state and type.

Reloading tables

To reload selected tables, select the tables you want to reload and then click the **Reload** button above the table list. When prompted to confirm the operation, click **OK**. The data in the selected tables will be reloaded to the target endpoint. Note that this option is not available for tasks that are defined *without* the Full Load replication option (such as "Apply Changes only" tasks).

Unsuspending tables

Tables that were suspended [due to errors](#) can be unsuspending once the errors are resolved. To unsuspend a table, simply select the table (or tables) and click the **Unsuspend** button.

i Tables in a Full Load and Apply Changes task that were suspended during the Full Load stage cannot be unsuspending. Such tables need to be reloaded instead.

It's possible to perform an unsuspend operation on multiple tables, but only if all tables are eligible for unsuspending. For example, simultaneously unsuspending a table that was suspended during the Full Load stage of a "Full Load and Apply Changes" task, and a table that was suspended during the Apply Changes stage, is not supported as the former table cannot be unsuspending.

i Changes that were applied to a table while it was suspended will be lost when that table is unsuspending. To work around this issue, simply define another task with the affected tables and use the "Start from timestamp" Run option to resume the task.

For information on the "Reload" and "Start from Timestamp Run" options, see *Working with tasks at runtime* (page 1203)

Information for tables that have completed loading

This section displays a table with information for each of the completed tables. To view this table, click the **Completed** bar, shown in the figure below.



Completed Tables Status

The information is displayed in a table that has the following columns:

- **Table name:** The names of the source tables that have completed loading.
- **Loaded On:** The time that Replicate started loading the table records to the target.
- **Transferred Count:** The number of records loaded to the target.



When [filters](#) are applied to source tables, the number shown in this column will reflect the total number of records before the include/exclude filter was applied. Consequently, the "Transferred Count" number may be higher than the actual number of records that were replicated to the target.

- **Transferred Volume:** The volume of the records including Replicate metadata (in MB) loaded to the target.



As the calculated target data volume includes the Replicate metadata (*table_id, stream_position, flags, bookmarks, and so on*), the source data volume will always be less than the target data volume.

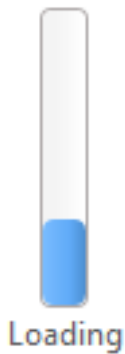
- **Load Duration:** The amount of time that it took for all records to load to the target.
- **Throughput Records:** The average throughput rate for the table. Throughput describes the number of records read per second.
For more information on throughput, see *Monitoring throughput in a Full Load operation (page 1224)*.
- **Throughput Volume:** The average throughput rate for the table. Throughput is the volume of records (in KB) read per second. Note that the transferred volume (see above) directly impacts the average throughput volume.
For more information on throughput, see *Monitoring throughput in a Full Load operation (page 1224)*.
- **Reload:** Click the **Reload** icon to reload the data for selected tables and run the full-load operation again.

Information for tables that are currently loading

This section displays a table with information for each of the tables that are currently loading. To view this table, click the **Loading** bar, shown in the figure below.



When replicating to an Oracle database with a full disk and/or partition where Oracle is trying to write archived redo log files, insert operations may fail. In such a case, no error will be shown and the task will not progress past the loading stage. To confirm that this is indeed an Oracle Archiver error, stop and attempt to restart the task. The task will not start and an appropriate error should be shown.



Loading Tables Status

The information is displayed in a table that has the following columns:

- **Table Name:** The names of the source tables that are currently loading.
- **Load Duration:** The amount of time that it took for all records to load to the current point in time.
- **Estimated Count:** The estimated number of rows that are to be loaded in the full load operation.
- **Transferred Count:** The number of records that are loaded to the target endpoint.



When [filters](#) are applied to source tables, the number shown in this column will reflect the total number of records before the include/exclude filter was applied. Consequently, the "Transferred Count" number may be higher than the actual number of records that were replicated to the target.

- **Current Throughput:** The current throughput rate for the table. Throughput describes the number of records read per second. For more information on throughput, see *Monitoring throughput in a Full Load operation* (page 1224).
- **Cached Changes** (not relevant for Full Load only tasks) - The number of changes (DMLs) applied to the source table since the Replicate started loading the table to the target.
- **Estimated Finish Time:** The approximate time the task finished loading the tables. The timestamp displayed indicates the date and time.

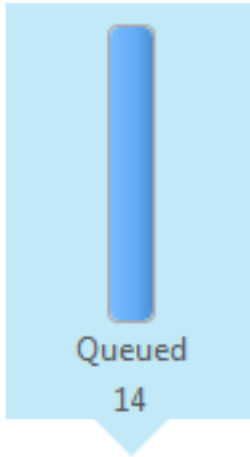


There may sometimes be a discrepancy between the "Estimated Finish Time" and the "Time Remaining (Estimated)" values. The "Time Remaining (Estimated)" value is calculated by the combined transfer rate of all the records of the task, while the "Estimated Finish Time" is calculated per table. The discrepancy arises when the table transfer rate at the beginning of the task is very fast, but slows down towards the end of the task. In this situation, the "Time Remaining (Estimated)" value will be greater and less accurate than the "Estimated Finish Time" value.

- **Progress:** The table status and the time the table entered that status.
- **Reload:** Click the **Reload** icon to reload the data for selected tables and run the full-load operation again.

Information for tables that are in the loading queue

This section displays a table with information for each of the tables that are waiting to be loaded. To view this table, click the **Queued** bar, shown in the figure below.



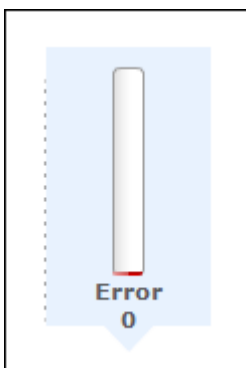
Queued Tables Status

The information is displayed in a table that has the following columns:

- **Table Name:** The names of the source tables that are currently in the queue waiting to be loaded.
- **Estimated Count:** The estimated number of rows that are waiting to be loaded in the full load operation.

Information for tables with errors

This section displays a table with information for each of the tables that stopped loading or suspended CDC due to an error. To view this table, click the Error bar, shown in the figure below.



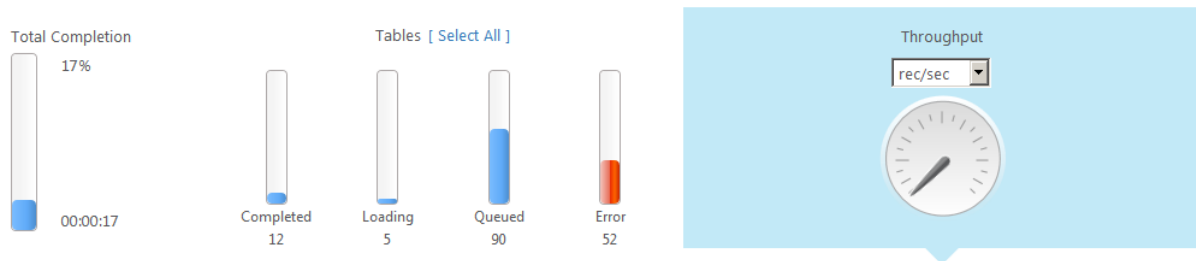
Error Tables Status

The information is displayed in a table that has the following columns:

- **Table Name:** The names of the source tables that stopped due to an error.
- **Failed On:** The time that the error occurred.
- **Loaded Count:** The number of records loaded when the error occurred.

Monitoring throughput in a Full Load operation

Throughput values for a full-load operation provide information on how fast the table records are being replicated to the target endpoint. The information is displayed in a gauge on the right side of the full-load graph section. The following figure shows the throughput gauge.



Throughput Gauge

You can set the throughput measurement values either to the number of records replicated per second, or to the number of kilobytes replicated per second. The display is always based on the current load operation.

To set the unit of throughput measurement:

- Select either **rec/sec** or **kbyte/sec** from the drop-down menu above the Throughput gauge.

Click the **Throughput** gauge to display a graph with the target throughput details. To view the graph only, click the expand/collapse arrow on the right of the gray bar above the graph. Click the arrow again to restore the status bars and throughput gauge.

15.3 Monitoring change processing operations

You can view the progress of the change-processing operation in the left section of the Monitor.

To make sure you are viewing the information for a change-processing operation, select the **Change Processing** tab.

You can view the following:

- *General Change Processing information (page 1224)*
- *Detailed Change Processing information (page 1226)*

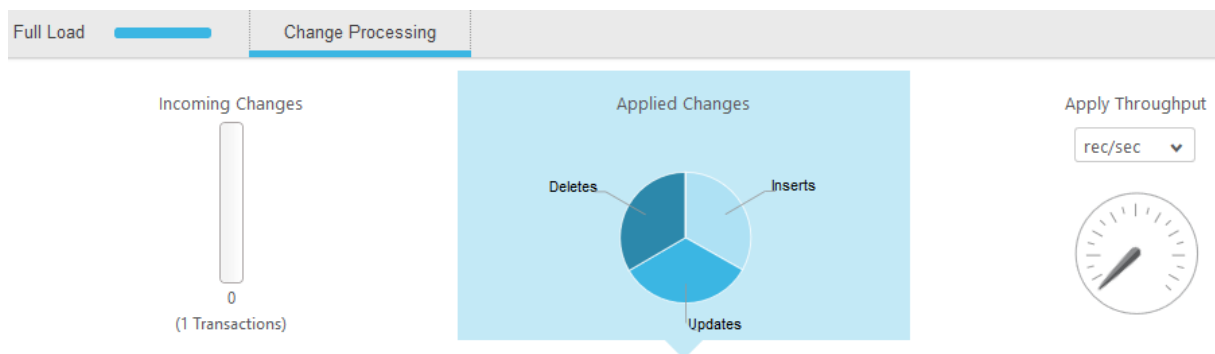
General Change Processing information

General information about the Change Processing task is presented in a graphical format. The following figure shows the graphical information displayed.



In an Apply Changes only task, Replicate performs the following operations depending on whether or not the target tables already exist:

- If a target table does not exist, Replicate will create it (metadata only).
- After the table is created, only INSERT operations will be supported. Other operations (e.g. UPDATE) will fail since the data does not exist (unless it has been inserted earlier).
- Stopping and resuming a task resets the values in the **Applied Changes** pie chart and in the **Applied Changed Details** columns below the chart.
- If the table already exists, Replicate will behave according to the [If target table already exists](#) setting in the task settings' **Full Load Settings** tab.



Change Processing Status

This section has the following information:

- **Incoming Changes:** The total number of records that were processed for the task.
- **Applied Changes:** A circle graph that shows information about the processed changes. It displays the following:
 - The number of INSERT operations processed. Hover over the Insert section with your mouse to see the number and percentage of the accumulated inserts.
 - The number of UPDATE operations processed. Hover over the Update section with your mouse to see the number and percentage of the accumulated updates.
 - The number of DELETE operations processed. Hover over the Delete section with your mouse to see the number and percentage of the accumulated deletes.
 - The number of metadata changes (DDL) processed. DDL changes include information about events like changes to table names or to column names.
- **Apply Throughput** gauge: A gauge that describes the number of change events read per second. For additional details, you can also view a graph with *Information about Change Processing throughput* (page 1230).
- **Apply Latency** gauge: A gauge that displays the latency information. The latency values displayed in the Qlik Replicate Console measure the time delay (latency) between the time when a change is visible to the source (and committed), and the time when this same change is visible to the target. The display is always based on the current change being applied. You should take the following into consideration:

- Latency when applying large transactions:
For example, when the most recent latency value was 10 seconds and now a transaction of one million rows gets committed at the source endpoint, Qlik Replicate starts to apply that transaction to the selected target and it will take some time to write all the changes to the target (for example 60 seconds). During the next 60 seconds, the latency value gradually grows to 70 seconds for the last change in the transaction. Once the transaction is committed, the latency drops back to the 'regular' latency (10 seconds in this case).
- Latency when no transactions are being applied:
When a time period passes with no changes applied to the target, the latency calculation is based on the time difference between the current time and the timestamp of the last change event read from the transaction log. This could happen if, for example, there is high activity on tables which are not selected for replication in the current task.

For additional details, you can also view a graph with *Information about Apply Latency* (page 1231).

Detailed Change Processing information

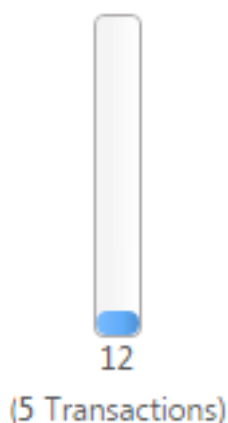
For each of the status indicators displayed in the *General Change Processing information* (page 1224) section, a table or graph is displayed in the section below with detailed information about the change processing status. The following information is available:

- *Information about Incoming Changes* (page 1226)
- *Information about applied changes* (page 1227)
- *Information about Change Processing throughput* (page 1230)
- *Information about Apply Latency* (page 1231)

Information about Incoming Changes

This section displays two bar graphs with information about incoming changes. Incoming changes displays a snapshot of the number of change records currently being read from the source endpoint and written to the target endpoint. To view these graphs, click the **Incoming Changes** bar, shown in the figure below.

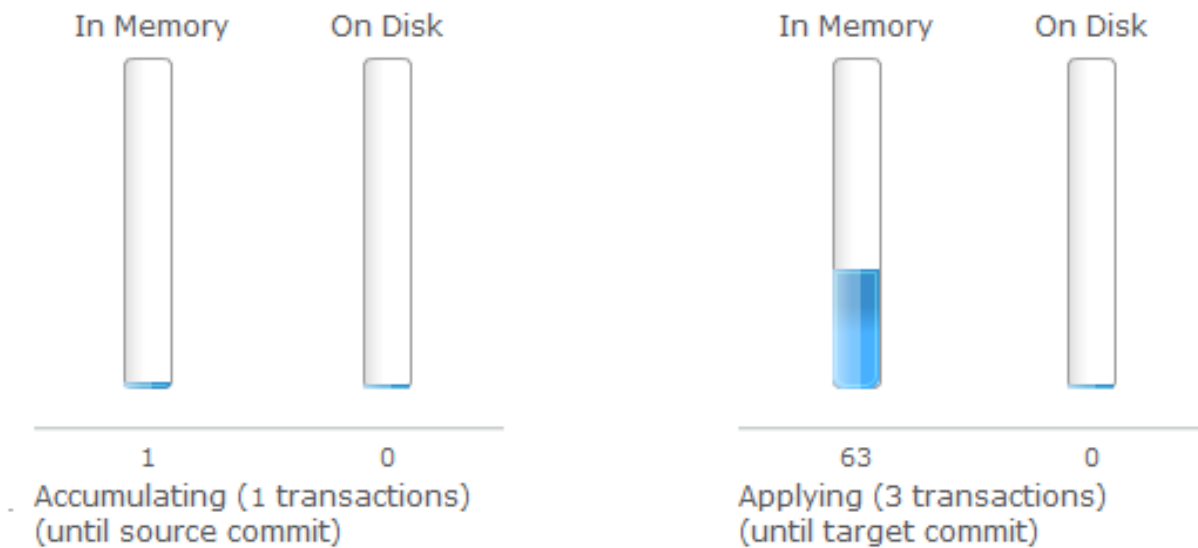
Incoming Changes



Incoming Changes

The following graphs are displayed.

Incoming Changes Details



Incoming Change Graphs

The graphs have the following information:

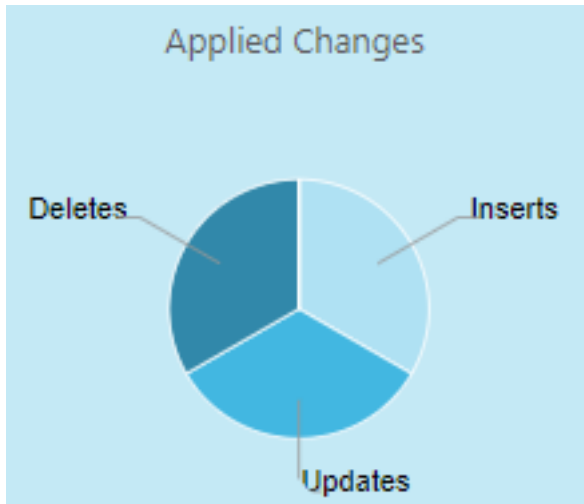
- **Accumulating:** These bars display the number of records currently being read from the source endpoint. These records are accumulated in a queue until they are applied to the target. The following is displayed:
 - **In Memory:** The number of accumulating records that are currently in the computer memory.
 - **On Disk:** The number of accumulating records that are currently stored on disk.
- **Applying:** The number of records currently being written to the target. These are the applied changes. The following is displayed:
 - **In Memory:** The number of records being applied that are currently in the computer memory.
 - **On Disk:** The number of records being applied that are currently stored on disk.

Information about applied changes

Information about the applied changes is shown in the following tables:

- *Recent Activity (page 1228)* - Shows changes that were committed to the source.
- *Aggregates (page 1229)* - Shows the total changes committed to the target for each change type and transaction type.

To view these tables, click the **Applied Changes** pie graph, shown in the figure below. The **Applied Changes** pie graph indicates which changes were committed to the target.



Applied Changes graph

The following tables are available when you click the **Applied Changes** pie chart:

- *Recent Activity* (page 1228)
- *Aggregates* (page 1229)

Recent Activity

Click the **Recent Activity** tab in the **Applied Changes Details** pane to view information about which changes occurred in each table, including:

- **Table Name:** The names of the source tables that are included in the task.
- **Insert:** The number of INSERT operations processed for the specific table.
- **Delete:** The number of DELETE operations processed for the specific table.
- **Update:** The number of UPDATE operations processed for the specific table.
- **DDL:** The number of metadata changes (DDL) processed. DDL changes include information about events like changes to table names or to column names.
- **Total Applied:** The total number of changes applied to the target.
- **Data Errors:** The number of data processing errors for the specific table. Data errors occur at the record level and include conversion errors, errors in transformations, and bad data.

Resetting the Data Errors Count

After you have resolved the data errors it is recommended to reset the data errors count. This is especially important if you have configured Replicate to perform an escalation action when the number of errors reaches a certain amount.

Details about the errors can be found in the `attrep_apply_exceptions` control table.

To reset the error count for a specific table, select the table and then click the **Reset data errors** button above the table list. Note that resetting the error count does not delete the error information from the `attrep_apply_exceptions` table.

For information about setting a data error escalation policy, see [Data Errors](#).

For information about the `attrep_apply_exceptions` table, see *Apply exceptions* (page 1298)



Reloading a table resets the data error count for that table.

- **Last Modified:** The time the last change occurred for the specific table.

Reloading or unsuspending tables

You can reload or unsuspend a table, depending on the current task state and type.

Reloading tables

To reload selected tables, select the tables you want to reload and then click the **Reload** button above the table list. When prompted to confirm the operation, click **OK**. The data in the selected tables will be reloaded to the target endpoint. Note that this option is not available for tasks that are defined *without* the Full Load replication option (such as "Apply Changes only" tasks).

Unsuspending tables

Tables that were suspended [due to errors](#) can be unsuspending once the errors are resolved. To unsuspend a table, simply select the table (or tables) and click the **Unsuspend** button.



Tables in a Full Load and Apply Changes task that were suspended during the Full Load stage cannot be unsuspending. Such tables need to be reloaded instead.

It's possible to perform an unsuspend operation on multiple tables, but only if all tables are eligible for unsuspending. For example, simultaneously unsuspending a table that was suspended during the Full Load stage of a "Full Load and Apply Changes" task, and a table that was suspended during the Apply Changes stage, is not supported as the former table cannot be unsuspending.



Changes that were applied to a table while it was suspended will be lost when that table is unsuspending. To work around this issue, simply define another task with the affected tables and use the "Start from timestamp" Run option to resume the task.

For information on the "Reload" and "Start from Timestamp Run" options, see *Working with tasks at runtime* (page 1203)

Aggregates

Click the **Aggregates** tab in the **Applied Changes Details** pane to view information about total changes for each change type and transaction type.

The Aggregate table displays the total changes (for all tables) applied for each of the following types of operations:

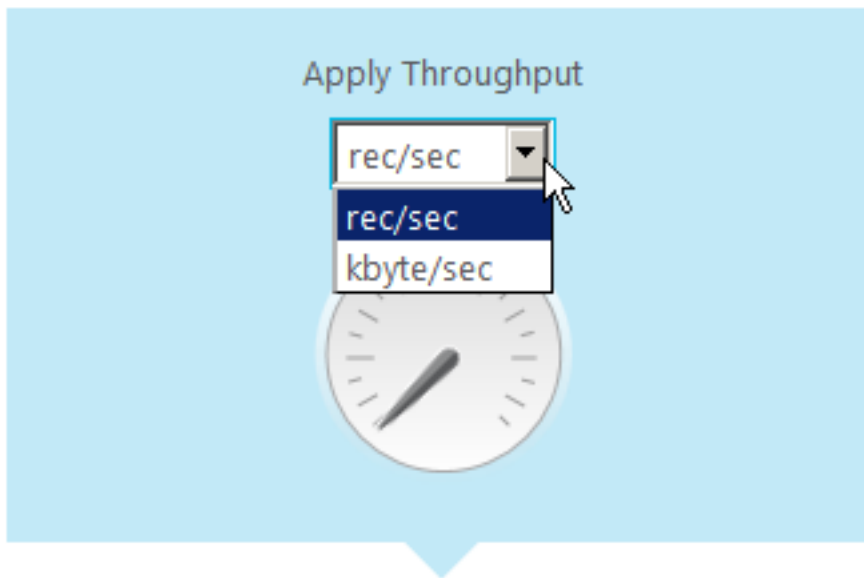
- INSERT
- UPDATE
- DELETE
- DDL

The Aggregate table also displays the information about transactions. It displays the total number and volume of:

- COMMITS
- ROLLBACKS

Information about Change Processing throughput

Throughput values for apply throughput in a change-processing operation provide information on how fast the change records are loaded to the target endpoint. The information is displayed in a gauge in the Change-Processing graph section. The following figure shows the Apply Throughput gauge:



Apply Throughput Gauge

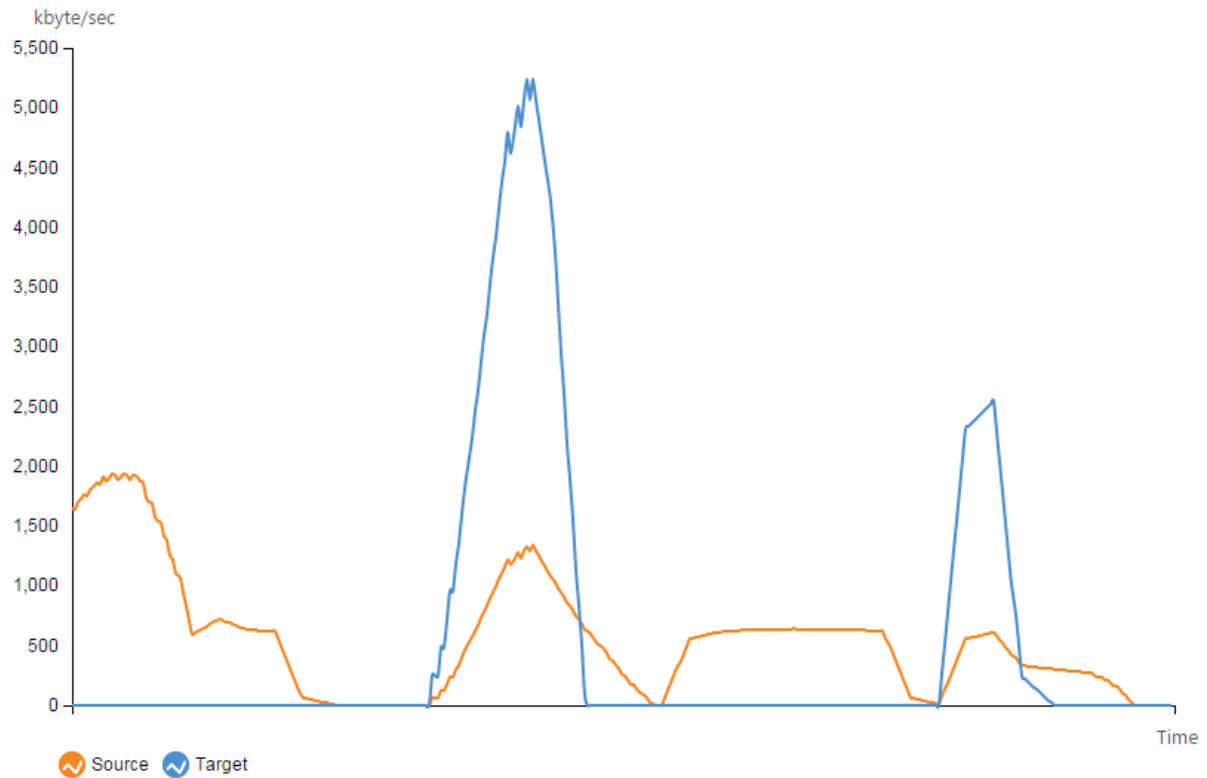
You can set the Apply Throughput measurement values either to the number of change records replicated per second, or to the number of kilobytes replicated per second. The display is always based on the current load operation.

To set the unit of throughput measurement:

- Select either **rec/sec** or **kbyte/sec** from the drop-down menu below the **Apply Throughput** gauge.

Click the **Apply Throughput** gauge to display a graph with the throughput details as shown in the figure below. To view the graph only, click the expand/collapse arrow in right side of the gray bar above the graph. Click the arrow again to restore the progress bars and Change Processing gauges.

Apply Throughput Details



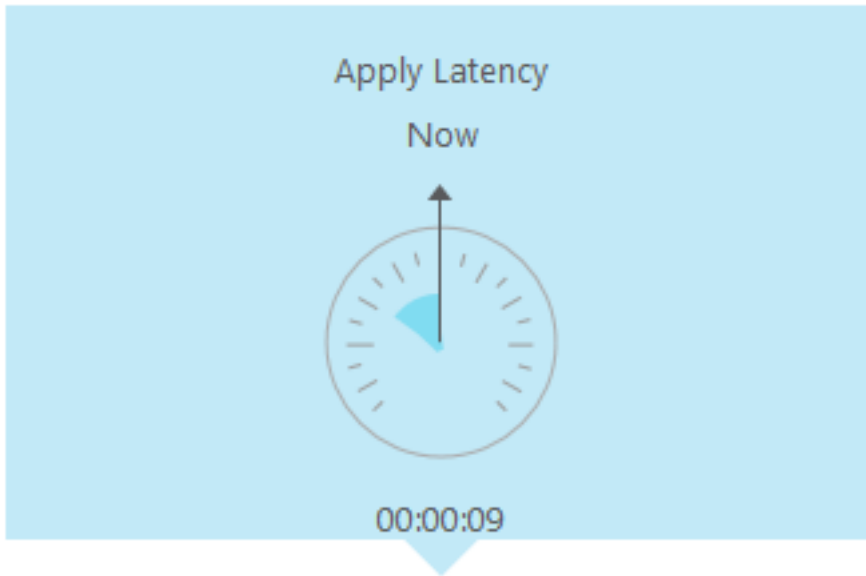
Apply Throughput Details Graph



The Apply Throughput metrics represented by the graph may not be always accurate. For accurate metrics, check the gauge described above.

Information about Apply Latency

Latency values for apply latency in a change-processing operation provide information about the time delay (latency) between the time when a change is visible to the source (and committed), and the time when this same change is visible to the target. The information is displayed in a gauge in the Change-Processing graph section. The following figure shows the Apply Latency gauge.



Apply Latency

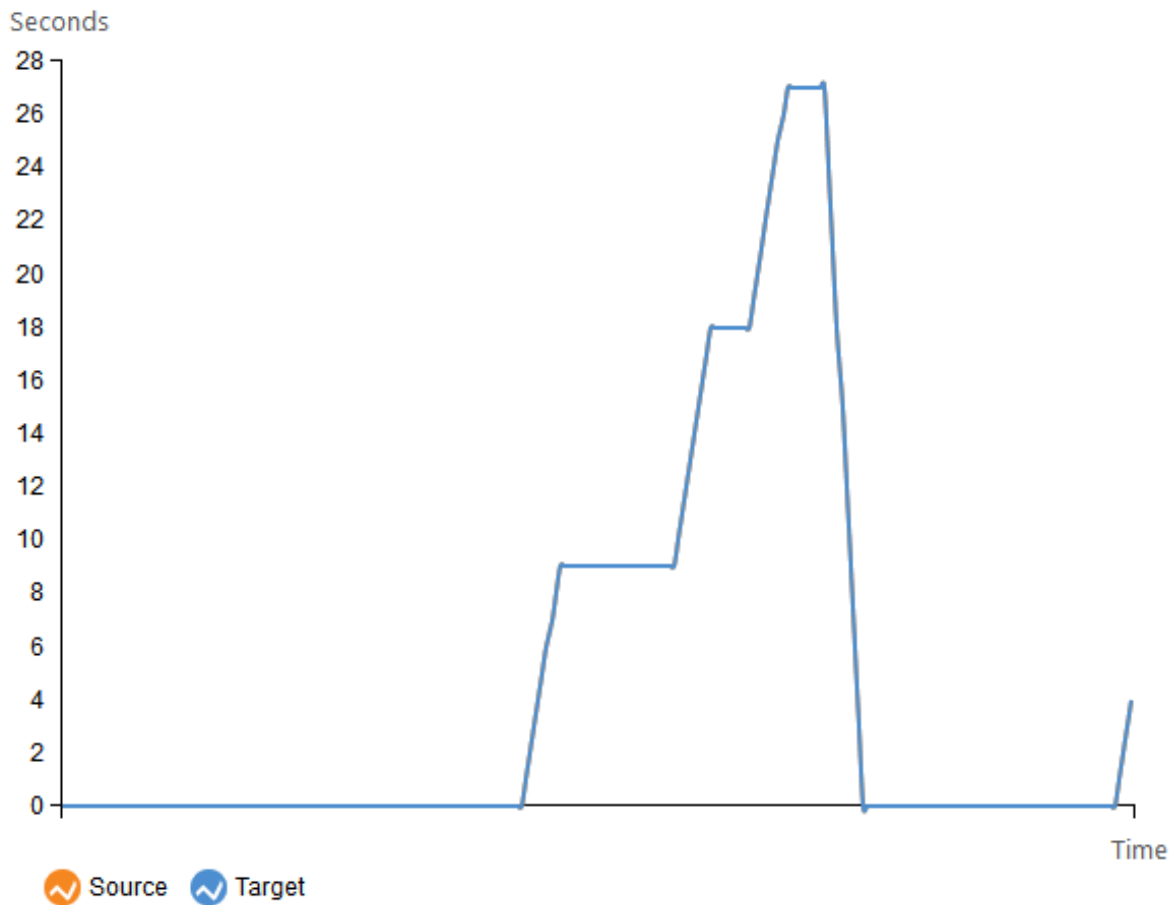
The latency values displayed in the Qlik Replicate Console measure the time delay (latency) between the time when a change is visible to the source (and committed), and the time when this same change is visible to the target. The display is always based on the current change being applied. For more information about latency, see [Apply Latency gauge](#).

Select the Apply Latency gauge to display a graph with the latency details. To view the graph only, click the expand/collapse arrow in right side of the gray bar above the graph. Click the arrow again to restore the progress bars and Change Processing gauges.



During data capture, the target latency will always be equal to the source latency (even though no data has reached the target yet). This is simply because target latency is the sum of source latency + apply latency, and can therefore never be less than the source latency.

Apply Latency Details



Apply Latency Details Graph

15.4 Viewing messages

You can see messages sent for the task while in the monitor view. For information on viewing messages, see *Accessing task messages and notifications* (page 1212).

15.5 Using the monitor tools

The monitor tools let you view additional information about the task. The following topics describe the information available through these tools:

- *Viewing history information* (page 1234)
- *Setting the task logging level* (page 1234)
- *Viewing, downloading, and manually rolling over task log files* (page 1235)
- *Deleting log files* (page 1236)
- *Downloading a diagnostics package* (page 1237)
- *Downloading a memory report* (page 1237)

Viewing history information

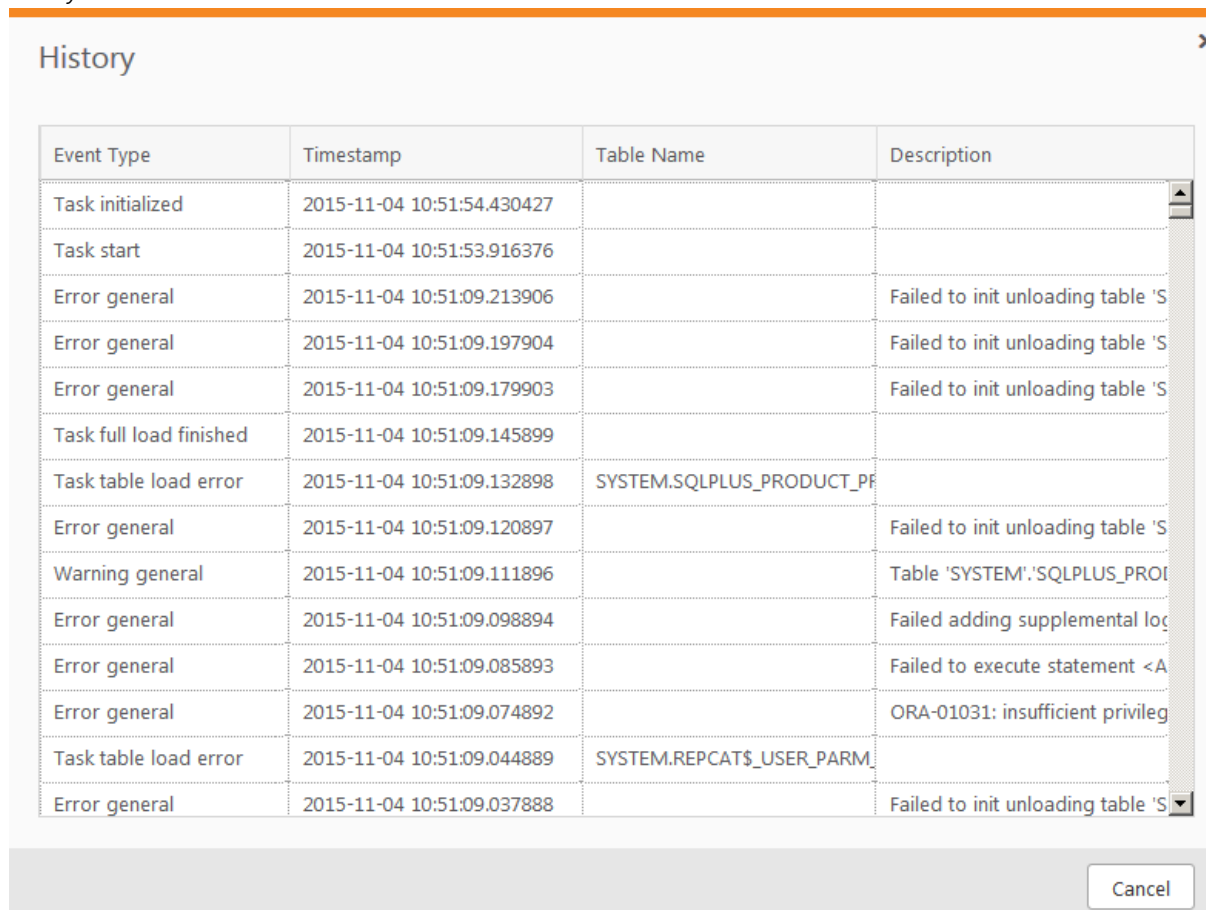
The **History** window displays information for each event carried out in the task. To access the **History** information, from *Monitor mode* (page 95), click **Tools** in the toolbar and then select **History**.

You can view the following information in the **History** window:

- **Event type:** The type of event that occurred, for example **Task started** or **Task table load finished**.
- **Timestamp:** A timestamp that indicates when the event took place. The timestamp is in the format, YYYY-MM-DD hh:mm:ss.milliseconds (to six places).
- **Table Name:** The name of the table where the event takes place if the event is related to a table.
- **Description:** A description of the event. This is not displayed for all events. You can double-click the description cell to view a window with the full message if the entire description is not available.

The following figure shows the History window.

History Window



Event Type	Timestamp	Table Name	Description
Task initialized	2015-11-04 10:51:54.430427		
Task start	2015-11-04 10:51:53.916376		
Error general	2015-11-04 10:51:09.213906		Failed to init unloading table 'S
Error general	2015-11-04 10:51:09.197904		Failed to init unloading table 'S
Error general	2015-11-04 10:51:09.179903		Failed to init unloading table 'S
Task full load finished	2015-11-04 10:51:09.145899		
Task table load error	2015-11-04 10:51:09.132898	SYSTEM.SQLPLUS_PRODUCT_PP	
Error general	2015-11-04 10:51:09.120897		Failed to init unloading table 'S
Warning general	2015-11-04 10:51:09.111896		Table 'SYSTEM':SQLPLUS_PROI
Error general	2015-11-04 10:51:09.098894		Failed adding supplemental log
Error general	2015-11-04 10:51:09.085893		Failed to execute statement <A
Error general	2015-11-04 10:51:09.074892		ORA-01031: insufficient privileg
Task table load error	2015-11-04 10:51:09.044889	SYSTEM.REPCAT\$_USER_PARM	
Error general	2015-11-04 10:51:09.037888		Failed to init unloading table 'S

Setting the task logging level

In the **Log Management** window, you can set the logging level for the task you are currently monitoring as well as view, download, and delete log files.



- The logging level can also be set in the **Logging Level** sub-tab in the **Task Settings** dialog box. For more information, see *Logging* (page 1192).

To set logging levels:

1. Open the task you are working with if it is not displayed in the Qlik Replicate Console. For information on opening a task, see *Editing a replication task* (page 125).
2. Switch to **Monitor** view. Then, click the **Tools** toolbar button and select **Log Management**. The **Log Management** window opens.
3. At the top of the **Log Management** window, set the **Component Logging Level** slider to the log level you want. This sets the logging level for all log modules. Note that all of the sliders for the individual modules move to the same position that you set in the main slider.
4. Make any changes to the sliders for the individual modules. This is optional. Note that if you change the main slider, all of the individual sliders are reset to the new position. If you want to maintain a different logging level for a specific module, you need to reset it.

Storing trace and verbose logging in memory

When the logging level is set to "Trace" or "Verbose", you can instruct Replicate to store the logging information in memory until an error occurs. On detecting an error, Replicate will begin writing to the physical logs and continue to do so for a few minutes after the initial occurrence of the error.

If *no* error occurs before the allocated memory is used up, Replicate will empty the memory buffer and start afresh.

This option is useful for tasks that fail unpredictably and for no obvious reason. The problem with continually writing large amounts of information to the logs is twofold:

- Running in "Trace" or "Verbose" logging mode will quickly use up available disk space (unless the logging settings have been configured to prevent this).
- Continually writing large amounts of data to the logs will affect performance.

To use this option

1. Select the **Store trace/verbose logging in memory, but if an error occurs, write to the logs** check box at the top of the tab.
2. In the **Allocate memory up to (MB)** field, specify the amount of memory you want to allocate for storing logging information.


Viewing, downloading, and manually rolling over task log files

In the **Log Viewer** window, you can view the logs for the task you are currently monitoring, manually roll them over if necessary, and download them.

Viewing and downloading the task log files

Follow the steps below to view or download the task log files.

To open the Log Viewer window:

1. Open the task whose log files you want to view or download.
For information on opening a task, see *Editing a replication task (page 125)*.
2. Switch to **Monitor** view.
3. Either, click the **Tools** toolbar button and then select **View Logs**.
OR
Click the **View Logs** button in the **Messages** pane in the lower right of the console.
The **Log Viewer** window opens.
4. Select the log file you want to view or download from the list in the **Log Files** pane. If you want to download the file, skip to [Step 8](#).
5. The contents of the log file will be displayed in the right pane. When you select a row in the log file, a tooltip will be display the full message of the selected row.
6. You can browse through the log file using the scroll bar on the right and the navigation buttons at the top of the window.
7. To search for a specific string in the log file, enter the search string in the search box at the top of the window.
Any terms that match the specified string will be highlighted blue.
8. To download the log file, click the  toolbar button.
Depending on your browser settings, one of the following will occur:
 - The task JSON file will be automatically downloaded to the default download location
 - You will be prompted for a download location. In this case, save the JSON file to your preferred location.

Manually rolling over task log files

You can manually roll the log file for the task you are monitoring in the Log Viewer. This lets you stop logging to the current log file and begin to log to a new log file. The current log file is called `reptask_<name of task>` and saved (older) log files have the file name `reptask_<name of task>_xxxxxxxxxxxx` where `xxxxxxxxxxxx` represents a 12-digit timestamp.

To immediately roll over the selected log file, click the **Roll Log File** button in the top right of the window.

Deleting log files

In the **Delete Logs** window, you can manually delete log files older than the specified number of days.

To delete the logs:

1. Open the task whose log files you want to delete.
For information on opening a task, see *Editing a replication task (page 125)*.
2. Switch to **Monitor** view.
3. Click the **Tools** toolbar button and then select **Delete Logs**.
The **Delete Logs** window opens.
4. Optionally change the default number of days (45) and then click the **Delete** button.
All log files older than the specified number of days will be deleted.

Deleting old partitions manually

Over time, Change Data Partitioning can result in a large number of files and partitions accumulating on the target system, which may significantly impact performance. Therefore, to ensure optimal performance, best practice is to delete old partitions from time to time.

Partitions can be deleted manually (as described below) or automatically as described in *Deleting old partitions periodically* (page 1178).

To manually delete processed partitions:

1. From the **Tools** menu, select **Delete Processed Changed Data Partitions**.
The Delete Processed Changed Data Partitions window opens.
2. From the **Delete partitions older than** drop-down lists, select a time period.
3. Click **Delete Partitions**.
4. If the task is running, the partitions will be deleted. Otherwise, a message will be shown, informing you that the partitions will only be deleted when the task starts running.



The minimum age of partitions that can be deleted may be preset by a consuming application. This is usually only happens when some partitions have not yet been processed. In this case, a note to this effect will be shown in the UI and you will not be able to change the minimum age.

Downloading a memory report

The memory report is a diagnostics tool that can be used to diagnose memory-related issues, such as unusually high memory consumption by a specific task.

Usually, multiple memory reports showing the gradual increase in memory consumption will need to be generated.

To download a memory report:

1. Open the task you are working with if it is not displayed in the Qlik Replicate Console. For information on opening a task, see *Editing a replication task* (page 125).
2. Click the **Tools** toolbar button and then select **Support > Download Memory Report**.
Depending on your browser settings, the following file will either be automatically downloaded to your designated download folder or you will be prompted to download it:
File name:
<task_name>__diagnostics__<timestamp>.memp
Example:
MyTask__diagnostics__20180109161333.memp
3. Send the report to Qlik.

Downloading a diagnostics package

You can generate a task-specific diagnostics package for Support to review. The diagnostics package contains the task log files and various debugging data that may assist in troubleshooting task-related issues.

To download a diagnostics package:

1. Open the task you are working with if it is not displayed in the Qlik Replicate Console. For information on opening a task, see *Editing a replication task (page 125)*.
2. Click the **Tools** toolbar button and then select **Support > Download Diagnostics Package**. Depending on your browser settings, the following file will either be automatically downloaded to your designated download folder or you will be prompted to download it:

File name:

<task_name>__diagnostics__<timestamp>.zip

Example:

MyTask__diagnostics__20180109161333.zip

16 Qlik Replicate server settings

This chapter describes how to configure the Qlik Replicate Server settings. Server settings are managed in **SERVER** view.

To switch to **SERVER** view:

- From the drop-down list in the top left corner of the console (below the product logo) select **Server**.



- *Server settings affect all Tasks that are created in the Qlik Replicate instance you are working with.*
- *Changes to server settings will not affect running tasks.*

In this section:

- *Notifications settings (page 1239)*
- *License settings (page 1260)*
- *Global error handling (page 1264)*
- *Logging (page 1265)*
- *File Transfer Service (page 1269)*
- *Endpoints (page 1278)*
- *Scheduling jobs (page 1272)*
- *User permissions (page 1274)*
- *Resource control (page 1276)*
- *External utilities (page 1277)*
- *More options (page 1282)*

16.1 Notifications settings

The following can be defined in the Notifications settings:

- *Defining notifications (page 1240)*
- *Setting up Mail parameters (page 1257)*
- *Creating a Default Recipient List (page 1258)*



Notifications will only be sent if the mail server settings are correctly configured.



Changes to notification settings will only take effect after restarting the specific tasks for which the notification is defined, or all tasks if the notification is defined for all tasks.

To view and edit the Notification settings:

- In **Server** view, click the **Notifications** tab on the left. Then click the **Notifications** sub-tabs to enter your settings.

Defining notifications

To configure and create notifications, click the **Notifications** sub-tab.

You use notifications to send messages about events that occur when running tasks in Qlik Replicate. Notifications are sent to inform users of any change in the system state, including:

- A task is started or stopped
- Latency is too high
- Memory utilization is too high
- Disk utilization is too high
- An error or a specific type of error occurred

You can manage notifications that you create from the Notifications list. This list provides you with information about each notification defined and lets you activate/deactivate a notification. In addition, you can make changes to the definitions of existing notifications or delete them.

The following topics describe how to define notifications in Qlik Replicate:

- *Creating a new notification (page 1240)*
- *The Notifications List (page 1256)*
- *Editing a notification (page 1256)*
- *Deleting a notification (page 1257)*

To open the Notifications page:

From the Server view, click **Notifications** from the menu list at the left. The **Notifications** sub-tab is displayed.

Notifications are sent by:

- An email message to the default list of users and/or to a custom list of users.
- Writing an entry in the Windows Event Log.
- Displaying a message in the Qlik Replicate Console.

Creating a new notification

Use the **New Notification** wizard to determine the notifications that are sent and who receives them.

To start the New Notification Wizard:

1. In Server view, click the **Notifications** tab on the left and then click the **New Notification** toolbar button.

2. From the drop-down menu, select **Task Events** or **Server Events** according to the notification you want to define.
3. The **New Notification** wizard opens displaying either the **Task Events** or **Server Events** screen (according to your earlier selection).
4. Continue from *Define the action that triggers the notification (page 1241)* or *Define the event that triggers the notification (page 1251)* as appropriate.

Creating a Notification for a task event

Use the **New Notification** wizard to create notifications for task-based events.



For changes to task notification settings to take effect, the task(s) needs to be stopped and resumed.

To create a notification for a task event

1. Launch the New Notification wizard as described in *Creating a new notification (page 1240)*.
2. In the **Notification Name** field, type a name for the notification.
3. Perform the following steps to define the notification:
 - *Define the action that triggers the notification (page 1241)*
 - *Define which changes of status trigger the notification (page 1242)*
 - *Define errors or warnings that trigger the notification (page 1244)*
 - *Define the recipients (page 1245)*
 - *Define the notification message (page 1245)*
 - *Associate tasks with the notification (page 1250)*
 - *Review the notification rule (page 1251)*

Define the action that triggers the notification

In the **Processing Status** section of the **Task Events** page, you can determine the action that triggers the notification. If the Operator section is not displayed, click on the header with the word Operator to display the options for this section. Select one of the following:

- **Task was started manually or by the Scheduler:** To send the notification when the task is started manually or by the Scheduler.
- **Task was stopped manually or by the Scheduler:** To send the notification when the task is stopped manually or by the Scheduler.
- **Task was stopped after Full Load: Cached changes were not applied:** To send the notification when the task is stopped after Full Load completes but before cached changes (changes to the source tables that occurred during Full Load) are applied to the target.
- **Task was stopped after Full Load: Cached changes were applied:** To send the notification when the task is stopped after Full Load completes and cached changes (changes to the source tables that occurred during Full Load) have been applied to the target.
- **Full Load started:** To send the notification when the Full Load process starts.
- **Full Load completed:** To send the notification when the Full Load process completes.

Once you determine when to send the notification, you can decide whether specific changes in status trigger the notification.

If you want to send a message about problems in latency, memory utilization, or disk utilization, click **Performance/Resources**. See *Define which changes of status trigger the notification (page 1242)* for an explanation.

If you want to send the notification when certain errors occur, click **Errors**. See *Define errors or warnings that trigger the notification (page 1244)* for an explanation.

Or you can click **Next** to *Define the recipients (page 1245)*.

Define which changes of status trigger the notification

In the **Performance/Resources** section of the **Task Events** page, you can define specific parameters for latency, disk utilization, or memory utilization that trigger a notification.

To set up notifications for latency, disk utilization, or memory utilization:

1. In the New Notification Wizard's **Task Events** page, click **Performance/Resources**.
2. Select one of the following:
 - **Latency is higher than [N] seconds.**
 - **Memory utilization exceeded [N] MB**
 - **Disk utilization exceeded [N] MB**
3. Define the value for the option you select. See the table below for an explanation on each of these options and how to set the value.



If you select one of these options, the notification is sent only when the selected parameter is true. However, you must also Define the action that triggers the notification (page 1241).

Set Values for Latency, Disk Utilization, Memory Utilization

Notification	Set Value	Notes
Latency is higher than Value seconds	Click [N] and enter a value in the field that is displayed. Latency is the time interval in seconds between the time a change was committed in the source system and the time it is applied and committed in the target system.	-

Notification	Set Value	Notes
Clear notification when latency drops below <n> seconds.	Use this to set the value that determines when latency returns to "normal limits." Click [N] and enter a value.	When latency is below the value entered in this field, it is considered to be in the "normal" range and the notification status ends. If selected, a notification is sent to indicate that latency returned to "normal" status. For more information, see <i>Define the notification message (page 1245)</i> .
Memory utilization exceeded Value MB	Click [N] and enter a value in the field that is displayed. Memory utilization is the amount of memory used by the task.	
Clear notification when memory utilization is below <n> MB	Use this to set the value that determines when memory utilization returns to "normal limits." Click [N] and enter a value.	When memory utilization is below the value entered in this field, it is considered to be in the "normal" range and the notification status ends. For more information, see <i>Define the notification message (page 1245)</i> .
Disk utilization exceeded Value MB	Click [N] and enter a value in the field that is displayed. Disk utilization is the amount of disk space used. Set a value that indicates that the current amount of disk space used is problematic to running a replication task.	
Clear notification when disk utilization is below <n> MB	Use this to set the value that determines when disk utilization returns to "normal limits." Click [N] and enter a value.	When disk utilization is below the value entered in this field, it is considered to be in the "normal" range and the notification status ends. For more information, see <i>Define the notification message (page 1245)</i> .

Once you determine the status changes that trigger a notification, you can decide whether specific errors trigger a notification.

If you want to send the notification when certain errors occur, click **Errors**. See *Define errors or warnings that trigger the notification (page 1244)* for an explanation.

Or you can click **Next** to *Define the recipients* (page 1245).

Define errors or warnings that trigger the notification

In the **Errors and Warnings** dialog of the **Task Events** wizard, you can determine whether notifications are sent when an error occurs. You can determine whether to send the notification for all errors or only for specific error types.

To set up notifications for errors:

1. In the New Notification Wizard's **Task Events** dialog, click **Errors and Warnings**.
2. Select one of the following:
 - **Task was stopped due to a non-recoverable error:** Select this to receive a notification when an error that cannot be retried is returned and the task is stopped due to this error.
 - **Table processing was suspended due to errors:** Select this to receive a notification when an error causes a table to stop processing during a full-load operation or suspend CDC. In this case, the table process stops, but the task continues.
 - **Any Error:** Select this to receive a notification when any error occurs in the system. Note that notification will not be sent for the following error types:
 - Recoverable errors. Recoverable errors are printed as INFO in the log (unless the task logging is set to TRACE, in which case they will be printed as ERROR).
 - Certain user-defined error handling policies. For example, if the **Duplicate key when applying INSERT** Apply Conflicts handling option is set to **Log record to the exceptions table**, no notification will be sent. However, if it set to **Stop task**, a notification *will* be sent.



Setting this option may result in a large volume of notifications, depending on the number of tasks running and their current status. It is therefore recommended to define this notification for selected tasks rather than for all tasks.

- **Any Warning:** Select this to receive a notification when a warning is issued in the system.
- **No changes were captured in the last <n> minutes:** Select this notification if you want to be alerted about delays in capturing changes from the source database. With most databases, receiving this notification will usually indicate a connectivity problem. However, with the following databases, it could simply mean that no changes occurred during the specified period.
 - Microsoft SQL Server
 - IBM DB2 for LUW
 - IBM DB2 for iSeries

When the task resumes capturing changes, a notification that change capture has resumed will be sent, and the notification timer will be reset.

Once you determine the error types that trigger a notification, you can:

- *Define the action that triggers the notification* (page 1241), if you have not done this already.
- *Define which changes of status trigger the notification* (page 1242) if you have not done this already.

- Or you can click **Next** to Define the Recipients.

Define the recipients

Notifications are always displayed in **Notifications** tab in the **Messages** panel of the Replicate Console. In the **Recipients** page of the New Notification wizard, you can determine whether to also send the notification to Windows Event Log and/or email recipients.

To determine the notification recipients:

Select any of the following to determine the notification recipients:

- **Event log:** Select this if you want the notification message to be written to the Windows/Linux Event log. For information on how to view the Windows/Linux Event log, see the online help for the version of Windows or Linux you are using.
- **Default notification email list:** Select this option if you want to send an email message to the all the recipients on the Default Notification Email List. For more information, see *Creating a Default Recipient List* (page 1258). To see the current list of default recipients, click **Show List**.
- **Custom email recipients:** Select this option to send the email notification to specific recipients.

Then:

- Click the **Add Recipient** button.
The **Name** field in the first available row in the list is activated.
- Type the name of the user that you want to receive the message.



If you click another part of the Replicate Console, the cell will become inactive. You can double-click the cell to enter additional information.

- Press the [tab] key or double click in the in the **Email** cell, then type the email address for the user you entered in the **Name** cell.
- Repeat the above to add more recipients.

When you have finished, click **Next** to Define the Notification Message.

Define the notification message

You can create a custom message for your notification. By default, a standard message is created based on your settings in the Task Events screen.

To create a notification message:

1. In the New Notification Wizard's, **Message** page, double-click in any of the table cells to open the Edit Notification Message dialog box. See the table below for an explanation of the information to enter in each field.
2. Click in the right pane of the dialog box and begin to type your message. In some cases a default message is displayed based on the information you entered in the previous pages of the New Notification Rule wizard. You can edit or delete the message, or create a new message to be sent with the notification in this dialog box.

3. Add variables in messages and email headers you define for notifications, if necessary. You can enter variables in one of two ways:
 - Type a variable into the message pane on the right using the following format:
`{{<VARIABLE_NAME >}}`
 For example: `{{TASK_NAME}}`.
 - Use the variables from the left pane of the **Edit Notification** dialog box. To add a variable to the notification message, you can:
 - Double-click the variable. The variable is inserted where your cursor is located in the notification message in the right pane.
 - Select the variable you want to use and click the arrow key in the middle of the **Edit Notification Message** dialog box. The variable is inserted where your cursor is located in the notification message in the right pane.
 - Drag the variable from the left pane to the location you want to place it in the notification message in the right pane.
 For more information, see the *Supported notification variables (page 1254)*.
4. Click **OK** to enter the message.

The following table describes how to enter the information in the **Message** page.

Creating a Notification Message

To where:	Notification On Message	Notification Off Message
<p>This column describes where the message is sent.</p> <p>For more information, see <i>Define the recipients (page 1245)</i>.</p>	<p>The Notification On Message is sent when the replication task meets the conditions for the notification to be sent.</p> <p>For more information, see <i>Define the action that triggers the notification (page 1241)</i>, <i>Define which changes of status trigger the notification (page 1242)</i>, and <i>Define errors or warnings that trigger the notification (page 1244)</i>.</p>	<p>The Notification Off Message is sent when the replication task returns to its normal state. This type of message is sent for notifications about latency, disk utilization, and memory utilization.</p> <p>For more information, see <i>Define which changes of status trigger the notification (page 1242)</i>.</p>

To where:	Notification On Message	Notification Off Message
<p>Console:</p> <p>The messages in this row are sent to the Qlik Replicate Console. They are displayed in the Messages section for a specific task. This section is displayed in:</p> <p>The Monitor for a specific task. For more information, see <i>Accessing task messages and notifications (page 1212)</i>.</p> <p>The right pane of the Tasks page. For more information, see <i>Viewing specific tasks (page 93)</i>.</p> <p>This message is also sent to the Windows Event log if you select this option. For more information, see <i>Define the recipients (page 1245)</i>.</p>	<p>In this field, you can edit, change or delete the message that is sent to the Qlik Replicate Console when the replication task meets the conditions for the notification to be sent.</p> <p>Example:</p> <pre>[{{SERVER_NAME}}\ {{NOTIFICATION_NAME}}] {{TASK_NAME}} replication task latency exceeds defined limits. Current latency is {{LATENCY}} seconds.</pre> <p>This message is sent to the console when latency reaches a value higher than the value you defined.</p>	<p>In this field, you can edit, change, or delete the message that is sent to the Qlik Replicate Console when the replication task returns to the normal range as you defined when you <i>Define which changes of status trigger the notification (page 1242)</i></p> <p>This field is relevant only for notifications about latency, disk utilization, and memory utilization.</p> <p>Example:</p> <pre>Latency is back to normal, latency is {{LATENCY}} seconds</pre> <p>This message is sent when latency returns to within its normal limits.</p>

To where:	Notification On Message	Notification Off Message
<p>Email Subject:</p> <p>This is the subject of the email messages sent for the notification.</p> <p>See <i>Define the recipients</i> (page 1245) for information about sending a notification as an email.</p>	<p>In this field, you can edit, change or delete the subject line for an email that is sent when the replication task meets the conditions for the notification to be sent.</p> <p>Example:</p> <pre>[{{SERVER_NAME}}\ {{NOTIFICATION_NAME}}] {{TASK_NAME}} high latency notification</pre> <p>This is the subject for an email message sent when latency reaches a value higher than the value you defined.</p>	<p>In this field, you can edit, change or delete the subject line for an email that is sent when the replication task returns to the normal range as you defined when you <i>Define which changes of status trigger the notification</i> (page 1242).</p> <p>This field is relevant only for notifications about latency, disk utilization, and memory utilization.</p> <p>Example:</p> <pre>Replicate notification ' {{NOTIFICATION_NAME}}' for task '{{TASK_NAME}}'</pre> <p>This is the subject for an email message sent when latency returns to within its normal limits.</p>

To where:	Notification On Message	Notification Off Message
<p>Email Message:</p> <p>This is the body of the email message sent for the notification.</p> <p>See <i>Define the recipients (page 1245)</i> for information about sending a notification as an email.</p>	<p>In this field, you can edit, change or delete the message that is sent by email when the replication task meets the conditions for the notification to be sent.</p> <p>Example:</p> <p>The latency for replication task {{TASK_NAME}} exceeds defined limits.</p> <p>The current latency is {{LATENCY}} seconds.</p> <p>----- ----- -----</p> <p>This is an automated message generated by Qlik Replicate server {{SERVER_NAME}} for notification {{NOTIFICATION_NAME}}.</p> <p>This is an email message sent when latency reaches a value higher than the value you defined.</p>	<p>In this field, you can edit, change, or delete the message that is sent by email when the replication task returns to the normal range as you defined when you <i>Define which changes of status trigger the notification (page 1242)</i>.</p> <p>This field is relevant only for notifications about latency, disk utilization, and memory utilization.</p> <p>Example</p> <p>Latency is back to normal, latency is {{LATENCY}} seconds</p> <p>This is an email message sent when latency returns to within its normal limits.</p>

To where:	Notification On Message	Notification Off Message
Event viewer	<p>In this field, you can edit, change or delete the message that is sent to the Windows/Linux event viewer when the replication task meets the conditions for the notification to be sent.</p> <p>Note: This field is available only when you select Event log when you <i>Define the recipients</i> (page 1245).</p> <p>Example:</p> <pre>[{{SERVER_NAME}}] \ {{NOTIFICATION_NAME}} {{TASK_NAME}} high latency notification</pre> <p>The latency for replication task {{TASK_NAME}} exceeds defined limits.</p> <p>The current latency is {{LATENCY}} seconds.</p> <p>This message is sent to the event viewer when latency reaches a value higher than the value you defined.</p>	-

After you define the message sent with the notification, click **Next** to associate tasks with the notification.

Associate tasks with the notification

By default, notifications are sent for all tasks that are defined in the Qlik Replicate instance you are using. You can determine whether to send the notification to specific tasks defined in the Qlik Replicate instance you are using. For example, you can define a different latency rate for a specific task that is replicating from a slow system.

To associate the notification with tasks:

- In the New Notification Wizard's **Associate** page, select one of the following:
 - All Tasks:** To associate this notification with all tasks that are defined in the Qlik Replicate instance you are working with. In this case all tasks that were previously defined and any future task will be associated with this notification.

If you choose to associate this notification with All Tasks, then click Next to review the notification rule.

- **Selected Tasks:** To associate this notification with one or more specific tasks only. Continue with the next step.
2. Select the check box next to any of the tasks you want to associate with this notification. You can select one or more tasks.



The Task check box at the top of the check-box column lets you select all of the tasks that are displayed. When you select this check box it is as if you select each of the tasks individually. Therefore, if you add tasks in the future they will not be included.

3. Click **Next** to Review the Notification Rule.

Review the notification rule

The **Summary** page lets you review the notification rule that you defined so that you can determine whether the selections you made in the wizard are correct. If you want to make changes, click **Back** and go to the page or pages you want to change.

When you are sure that the notification rule is defined in the way that you want, click **Finish** to close the wizard and add the rule to the notification list (see *The Notifications List* (page 1256)).

After you close the wizard, make sure to click **Save** at the top of the **Settings** page. This will save the information for all settings, not only for the notification rule that you created. If you made changes that you do not want to keep, click **Discard** to discard all changes before you make changes to any of the other settings.

Creating a notification for a server event

Use the **New Notification** wizard to create notifications for server-based events.



*Server notifications are not written to the console (**Task > Monitor > Notifications**).*

To create a notification for a server event

1. Launch the New Notification wizard as described in *Creating a new notification* (page 1240).
2. In the **Notification Name** field, type a name for the notification.
3. Perform the following steps to define the notification:
 - a. *Define the event that triggers the notification* (page 1251)
 - b. *Define the recipients* (page 1253)
 - c. *Define the notification message* (page 1253)
 - d. *Review the notification rule* (page 1254)

Define the event that triggers the notification

In the **Disk Space** section, you can determine the disk space utilization event that triggers the notification.



Disk space is checked only for the drive where the data folder resides.

Select one of the following:

- **Disk space utilization reaches the high threshold:** The notification will be triggered when disk space utilization reaches the percentage defined for the high threshold.
- **Disk space utilization reaches the critical threshold:** The notification will be triggered when disk space utilization reaches the percentage defined for the critical threshold.
- **Disk space utilization returns to normal:** The notification will be triggered when disk space utilization returns to normal percentage (i.e. not high or critical).
- **Disk space utilization reaches any of the defined thresholds or returns to normal:** The notification will be triggered in any of the following scenarios:
 - Disk space utilization increases from normal to the high threshold
 - Disk space utilization increases from normal to the critical threshold
 - Disk space utilization increases from the high threshold to the critical threshold
 - Disk space utilization returns to normal from the high threshold
 - Disk space utilization returns to normal from the critical threshold
 - Disk space utilization returns to the high threshold from the critical threshold

In the **System Memory** section, you can determine the system memory utilization event that triggers the notification.



Memory utilization check also takes into account the swap file space.

Select one of the following:

- **System memory utilization reaches the high threshold:** The notification will be triggered when system memory utilization reaches the percentage defined for the high threshold.
- **System memory utilization reaches the critical threshold:** The notification will be triggered when system memory utilization reaches the percentage defined for the critical threshold.
- **System memory utilization returns to normal:** The notification will be triggered when system memory utilization returns to normal percentage (i.e. not high or critical).
- **System memory utilization reaches any of the defined thresholds or returns to normal:** The notification will be triggered in any of the following scenarios:
 - System memory utilization increases from normal to the high threshold
 - System memory utilization increases from normal to the critical threshold
 - System memory utilization increases from the high threshold to the critical threshold
 - System memory utilization returns to normal from the high threshold
 - System memory utilization returns to normal from the critical threshold
 - System memory utilization returns to the high threshold from the critical threshold

Click **Next** to *Define the recipients* (page 1253).

Define the recipients

For more information, see *Define the recipients (page 1245)*.

Define the notification message

You can create a custom message for your notification. By default, a standard message is created based on your settings in the **Server Events** screen.


To edit the default notification message:

1. In the [New Notification Wizard](#), **Message** page, click the message text to open the **Edit Notification Message** dialog box. See the table *Define the event that triggers the notification (page 1251)* for an explanation of the information to enter in each field.
2. Click in the right pane of the dialog box and begin to type your message. In some cases a default message is displayed based on the information you entered in the previous pages of the New Notification Rule wizard. You can edit the message or create a new message to be sent with the notification in this dialog box.
3. Optionally, add variables in messages and email headers you define for notifications. You can enter variables in one of two ways:
 - Type a variable into the message pane on the right using the following format:
`{{<VARIABLE_NAME >}}`
 For example: `{{TASK_NAME}}`.
 - Use the variables from the left pane of the **Edit Notification Message** dialog box. To add a variable to the notification message, you can:
 - Double-click the variable. The variable is inserted where your cursor is located in the notification message in the right pane.
 - Select the variable you want to use and click the arrow key in the middle of the **Edit Notification Message** dialog box. The variable is inserted where your cursor is located in the notification message in the right pane.
 - Drag the variable from the left pane to the location you want to place it in the notification message in the right pane.
 For more information, see the *Supported notification variables (page 1254)*.
4. Click **OK** to enter the message.
5. After you define the message sent with the notification, click **Next** to *Review the notification rule (page 1254)*.

The following table describes how to enter the information in the **Message** page.

Message page entry fields

To where:	Notification Message
This column describes where the message is sent. For more information, see <i>Define the recipients (page 1253)</i> .	Sent when the server meets the conditions for the notification to be sent.

To where:	Notification Message
<p>Console:</p> <p>The messages in this row are sent to the Console.</p> <div data-bbox="204 416 823 663" style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">  <p><i>The same message is also sent to the Windows Event Log if you chose to send messages to the Windows Event Log. For more information, see Define the recipients (page 1253).</i></p> </div>	<p>In this field, you can edit the message that is sent to the Qlik Replicate Console when the server meets the conditions for the notification to be sent.</p>
<p>Email Subject:</p> <p>This is the subject of the email messages sent for the notification.</p> <p>See Define the recipients (page 1253) for information about sending a notification as an email.</p>	<p>In this field, you can edit the subject line for an email that is sent when the server meets the conditions for the notification to be sent.</p>
<p>Email Message:</p> <p>This is the body of the email message sent for the notification.</p> <p>See Define the recipients (page 1253) for information about sending a notification as an email.</p>	<p>In this field, you can edit the message that is sent by email when the server meets the conditions for the notification to be sent.</p>

Review the notification rule


The **Summary** page lets you review the notification rule that you defined so that you can determine whether the selections you made in the wizard are correct. If you want to make changes, click **Back** and go to the page or pages you want to change.

When you are sure that the notification rule is defined in the way that you want, click **Finish** to close the wizard and add the rule to the notification list (see [The Notifications List](#) (page 1256)).

After you close the wizard, make sure to click **Save** at the top of the **Settings** page. This will save the information for all settings, not only for the notification rule that you created. If you made changes that you do not want to keep, click **Discard** to discard all changes before you make changes to any of the other settings.

Supported notification variables

The table below describes which variables can be included in notification messages.



The following variables are not supported even though they are available for selection: ACTIVE_TABLES, ERROR_TABLES, COMMAND, COMMAND_CODE, FULLLOAD_ERROR_TABLES, ERROR_CODE, SQL_STATEMENT, and COMMAND_PARAMETERS.

Notification variables

Variable	Description
LATENCY	The task latency. For a definition of latency, see <i>Terminology (page 32)</i> .
MEMORY_USAGE	The amount of memory being consumed by all tasks on the Replicate Server machine.
DISK_USAGE	The amount of disk space being utilized by the task on the Replicate Server machine.
COUNT_ERROR_TABLES	The number of tables in the task with an error status.
COUNT_ACTIVE_TRANSACTION	The number of open transactions in the task.
COUNT_DATA_ERRORS	The number of data errors encountered by the task. For more information on data errors, see <i>Global error handling (page 1264)</i>
LOADED_RECORDS	The number of records loaded to the target database during the task.
CHANGES_RECORDS	The number of change records processed during the task.
FULLLOAD_COUNT_REQUESTED_TABLES	The number of tables that are queued for loading to the target.
FULLLOAD_COUNT_COMPLETED_TABLES	The number of tables that were loaded to the target.
FULLLOAD_COUNT_ERROR_TABLES	The number of tables that could not be loaded to the target due to error.
FULLLOAD_REQUESTED_TABLES_LIST	The names of the tables that are queued for loading to the target.
FULLLOAD_COMPLETED_TABLES_LIST	The names of the tables that were loaded to the target.
TABLE_NAME	The name of the table being processed when the notification was sent.
TABLE_OWNER	The owner of the table being processed when the notification was sent.
RECORD_COUNTER	The number of records that had been processed when the notification was sent.
ERROR_TEXT	The error message when a task ends with an error.
TASK_NAME	The name of the task.
NOTIFICATION_NAME	The name of the notification.

Variable	Description
TABLE_COUNT_APPLY_ERRORS	The number of tables with Apply Conflict errors. For more information on Apply Conflict errors, see <i>Global error handling (page 1264)</i>
SERVER_NAME	The host name of the Replicate Server machine.
STORAGE_UTILIZATION_OLD_STATE	The storage utilization state before it moved to the "new" state described below. For a description of possible states, see <i>Define the event that triggers the notification (page 1251)</i> .
STORAGE_UTILIZATION_NEW_STATE	The storage utilization state after it moved from the "old" state described above. For a description of possible states, see <i>Define the event that triggers the notification (page 1251)</i> .
USED_STORAGE_UTILIZATION	The amount of disk space (on the drive where the Replicate Data folder is located) used by all tasks.
TOTAL_STORAGE_UTILIZATION	The total amount of disk space available on the drive where the Replicate Data folder is located.
TIME_SINCE_LAST_CHANGE	The time that has elapsed in minutes since changes were last captured from the source database.

The Notifications List

The Notifications List lists all of the notification rules that are defined for the Qlik Replicate instance you are working with. It has the following information:

- **Name:** Displays the name of the notification rule.
- **Condition:** Displays the condition that triggers the notification to be sent. For more information, see *Creating a new notification (page 1240)*.
- **Send Message To:** Displays custom users that receive the message. For more information, see *Creating a new notification (page 1240)*.
- **Tasks:** Displays the tasks that are associated with this notification rule. For more information, see *Creating a new notification (page 1240)*.
- **Active:** Select the check box in this column to activate the notification. If this check box is cleared, notifications defined by this rule are not sent. This check box is selected by default.

Editing a notification

You can make changes to any notification rule.

To edit a notification rule:

1. From the [Notification List](#), select the notification you want to edit.
2. Click **Open** (at the top of the list).
or
Double-click the notification you want to edit.
The Edit Notification Rule wizard opens.
3. Make any changes you need in the wizard. For information on how to work with each of the pages in the New Notification Rule wizard, see *Creating a new notification (page 1240)*.



You can only make changes to those sections that you defined when Creating a new notification (page 1240).

- *You cannot change name of the notification.*
- *If you defined a notification to let you know when the task or full load started or stopped, this cannot be edited. For example, if you created a notification rule for starting a task and you now also want to get notified when the task stops, you must create a new notification rule.*
- *In the Notify When? page, you can make changes to the data you defined in the original notification rule. For example, if you defined a **Memory utilization** message in the **Notify when?** page, **Performance/Resources** section, you can only change this parameter. If you want to add information about something that was not defined in the original notification rule, for example you want to add errors to your notification or you want to get information about latency, you must create a new notification rule.*

Deleting a notification

You can delete notification rules that you no longer want to use.



When you delete a notification, it is deleted permanently.

To delete a notification:

1. From the [Notification List](#) select the notification you want to delete.
2. Click **Delete** (at the top of the list).

Setting up Mail parameters

The Mail parameters define the mail server used to send notifications.

To set the Mail parameters:

1. Click the **Mail Settings** sub-tab and enter the following information:
 - **Mail server:** Specify the outgoing mail server you wish to use to send task and server notifications, for example, `smtp.example.com`.

- **Port:** Specify the port number where the mail server is located. The default value is **25**.
 - **Use secure email (SMTPS):** Select to connect to the mail server (for sending notifications) using SMTPS.
 - **Verify host:** Select to verify that the host name in the certificate matches the host name of the mail server.
 - **Verify peer:** Select if you want the mail server to send a certificate proving its identity. On Windows, the certificate will be authenticated using the default CA certificates.
 - **CA Path:** On Linux, specify the location of the server CA certificate.
 - **Anonymous login:** Select to allow a Qlik Replicate user to access the mail server without having to provide any user credentials.
 - **User name:** Specify the user name for logging in to the SMTP Server.
 - **Password:** Specify the password for logging in to the SMTP Server.
 - **Sender email address:** Enter the email address that sends the email notifications. This is the address that appears in the **From** field of the email notification.
 - **Send test email:** Click to open the **Send Test Email** dialog box.
 - **Email address for test email:** Type an email address to receive a test email message from the server you configured. Use this to determine that the **Mail Parameters** you defined are valid.
2. Click the **Save** toolbar button to save all of your changes.



*If you made changes that you do not want to keep, click **Discard** to discard all changes before you make changes to any of the other settings.*

Creating a Default Recipient List

Click the **Default Recipients List** sub-tab to create a default recipient list.

A default recipient list is a list of recipients that receive all of the notifications that you define for task-based or server-based events. This allows you to use one list for all email notifications without having to define the list each time you create a notification.



You can choose to send notifications to a different list or to additional users for any specific notification. You define these exceptions when you create the specific notification. For more information, see [Creating a new notification \(page 1240\)](#).

To create a Default Recipient List:

1. At the top of the **Default Recipient List** settings page, click **Add Recipient**.
The next row in the Recipient List table becomes available.
2. Type the name of the user you want to add to the list of default recipients. Continue to enter a name and email address for each recipient you want to include in the default list.
3. Press the [tab] key or double click in the in the **Email** cell, then type the email address for the user you entered in the **Name** cell.
4. Click **Save** at the top of the screen to save all of the changes you made.



Click **Save** to save the information for all settings, not only for the recipient list. If you made changes that you do not want to keep, click **Discard** to discard all changes before you make changes to any of the other settings.

Replicate Event IDs in Windows Event Log

The table below lists the Event IDs for Replicate events in Windows Event Log.



Some events share the same ID. With these events, the recommended way of identifying the event is to parse the notification subject. This is especially recommended if you are using third-party tools to detect and report events.

Other Event IDs

Event ID	Description
261	Any error.
262	Any warning.
263	Errors containing the specified codes.

Server Event IDs

Event ID	Description
300	Server disk usage has changed.
300	Disk utilization has changed
300	Server disk usage has reached the high threshold.
300	Server disk usage has reached the critical threshold.
300	Server disk usage has returned to normal.
320	Server system memory usage has changed.
320	Memory utilization has changed.
320	Server system memory usage has reached the high threshold.
320	Server system memory usage has reached the critical threshold.
320	Server system memory usage has returned to normal.
340	Server state has changed.

Task Event IDs

Event ID	Description
400	Task has started.

401	Full load has started.
402	Full load has completed.
403	Task has stopped after Full Load – cached changes were applied.
404	Task has stopped after Full Load – cached changes were not applied.
405	Task has stopped (other cases).
406	Task has stopped due to a non-recoverable error.
261	Task latency has exceeded the set limit.
261	Task latency is back to normal.
432	Task memory usage has exceeded the set limit.
433	Task memory usage is back to normal.
434	Task disk usage has exceeded the set limit.
435	Task disk usage is back to normal.

Table Event IDs

Event ID	Description
502	Table processing was suspended due to errors.

16.2 License settings

You need to register the software before you can use Qlik Replicate. Your Qlik vendor should provide you with a text file called `license.txt`. This file contains details such as the product expiration date (if any).

Use the License settings page for:

- *Requesting a license (page 1260)*
- *Registering a license (page 1263)*
- *Viewing a license (page 1264)*



Changes to settings will only take affect after restarting all tasks.

To open the License settings page:

- From the Server view, click **License** from the menu list at the left. The **License** sub-tab is displayed.

Requesting a license

You must have a valid license to work with Qlik Replicate. You can request a license from the License settings page in the Qlik Replicate Console. In the License Request dialog box, fill out the required information and submit the request by email. Once your request is approved, the license file is sent to you by email. To use Qlik Replicate, register the license by using the procedure described in *Registering a license (page 1263)*.

To request a license:

1. From the **Server** page, click **License**.
2. At the top of the **License** tab, click **Request License**.
The **Replication License Request** dialog box opens.

Qlik Replicate License Request ✕

Request type ▾

License to

License type ▾

Expiration date

Hosts

Source types

Target types

Advanced...

Send by Mail

3. Enter the requested information:
 - Request type: Select one of the following:
 - **New License:** Select this if this is your initial license request.
 - **Extend License:** Select this if you have a license and want to extend its period of validity.
 - **Alter License:** Select this if you want to make changes to an existing license. For example, if you want to add additional sources or targets or change the host computer.

- **License to:** Type the name of the company or group that is requesting a license to use Qlik Replicate.
 - **License type:** Select one of the following:
 - **Permanent:** Select this if the license will always be valid. Permanent licenses do not require an expiration date.
 - **Evaluation:** Select this if you are requesting a temporary license to use Qlik Replicate for a trial period.
 - **Term:** Select this if you are requesting a license that is valid for a specific period of time. In this case you must be sure to include an expiration date in your request.
 - **Expiration date:** Click in this field to select the expiration date using the pop-up calendar. This is required only if you selected **Evaluation** or **Term** in as the License type.
 - **Hosts:** Type the name of the local computer where Qlik Replicate is installed. By default the name of the local computer is displayed in this field. You can change this or add additional computers if you are installing Qlik Replicate in a different or an additional location.
 - **Source Types:** Click **Edit** to open the Edit Source Types dialog box. Check the endpoint types you are working with as your replication sources. You can select one or more endpoint endpoints as necessary. If you need to work with all available endpoints, click **All**.
 - **Target Types:** Click **Edit** to open the Edit Target Types dialog box. Check the endpoint types you are working with as your replication targets. You can select one or more endpoint endpoints as necessary. If you need to work with all available endpoints, click **All**.
4. Click **Send by Mail** to open an email request for the license. Send the email to the address entered in the recipient field of your default email client.



*If you have not registered a default email client, clicking the **Send by Mail** button will not open your email client. For instructions on registering a default email client, refer to your browser's or operating system's online help.*

Click **Copy to Clipboard** to copy the information to the computer's clipboard. You can paste this information into the Advanced license request and edit it as necessary. For more information, see **Using the advanced license request option**.

Using the advanced license request option

The advanced license request option lets you request a license by manually typing the information. Make sure to include all of the information required as described in **Requesting a license**. The following is a suggested format for the advanced option:

```
Request type:New License
License to: <company name>
License type: Permanent
Expiration date:
Hosts: bee01-xp.company.local
Source Types: Oracle
Target Types: SQLServer
```

Registering a license

You must have a valid license to work with Qlik Replicate. If you did not receive a license.txt file, you can request a license using the procedure described in *Requesting a license (page 1260)*. Once you receive the license, you must register it to work with Qlik Replicate.

To register a license:

1. Copy the license.txt file to your computer or any computer in your network you have access to.
2. From the Server page, click **License**.
3. At the top of the **License** tab, click **Register License**.

The **Register License** dialog box opens.

Register License

Load license from file or copy & paste to screen

license.txt Load...

License text:

```
#  
# Temporary license for development  
#  
license_type=evaluation_license  
licensed_to=Attunity Internal - Attunity Replicate Development  
licensed_by= Repli_man  
serial_no=212337  
expiration_date=2012-08-31  
source_types=  
target_types=  
features=  
version=2.0.0.0  
issue_date=2012-07-29  
checksum=JE9EF-CHHFH-79R24-F42K2
```

To register new or modified license click Register License.

Register License Close

4. Click **Load** and browse to find and select the license file.

The license text is displayed in the dialog box as shown above. Check to be sure that the details are correct.

5. Click **Register License** to register the license. A message indicating the license was registered successfully is displayed.



A message is displayed at the top of the Qlik Replicate Console that indicates that you have a valid license and when it expires. If the license is expired or invalid, the message indicates this.

You can also click on this message link to request, register, or view license information.

Viewing a license

You can view the license information in the Qlik Replicate Console at any time.

To view the license information:

- From the Server page, click **License**.

The **License** tab is displayed. All of the license information is displayed in the **License** tab.

Registered License	
Checksum	R8H65-8J933-5R5E3-9C933
Expiration Date	2012-05-31
Features	
Hosts	
issue_date	2012-05-13
License Type	EVALUATION_LICENSE
License By	Repli_man
License To	Attunity Internal - Attunity Replicate Development
Serial Number	212334
Source Types	
Target Types	
Version	2.0.0.0

16.3 Global error handling

You can configure how Qlik Replicate responds to specific types of errors. You can define error handling on the task level or the server level. The configurations you make in the Server Settings affect all tasks created for

this instance of Qlik Replicate unless you define a task to use the definitions you create for that task. For information on how to configure error handling for a specific task, see *Error handling (page 1186)* in the *Customizing tasks (page 1082)* chapter.



- *Changes to settings will only take affect after restarting all tasks.*
- *Global error handling is not available when the [Apply changes using SQL MERGE](#) task setting is enabled.*

In Server view, click the **Global Error Handling** tab on the left.

The following sub-tabs are available:

- **Environmental Errors:** An error that is caused by an environmental problem in the source or target endpoint or on the network. Environmental errors can be restarted.
The information you enter in this tab is the same as the information you enter in the **Environmental Errors** tab for tasks. For information about the options available in this tab, see *Environmental errors (page 1187)*.
- **Data Error:** An error related to data processing at the record level.
The information you enter in this tab is the same as the information you enter in the **Data Error** tab for tasks. For information about the options available in this tab, see *Data Errors (page 1188)* in the Customizing Tasks chapter.
- **Table Error:** An error in processing data or metadata for a specific table. This only includes general table data and not an error that relates to a specific record.
The information you enter in this tab is the same as the information you enter in the **Table Error** tab for tasks. For information about the options available in this tab, see *Table Errors (page 1191)* in the Customizing Tasks chapter.
- **Apply Conflicts:** Errors that occur when the target endpoint is not synchronized with the source endpoint when processing changes. This can cause duplicate key errors on INSERT operations or zero rows affected on UPDATE/DELETE operations.
The information you enter in this tab is the same as the information you enter in the **Apply Conflicts** tab for tasks. For information about the options available in this tab, see *Apply Conflicts (page 1191)* in the Customizing Tasks chapter.

16.4 Logging

The following topics describe the server logging management options:

- *Setting logging levels for the Server and File Transfer Service (page 1266)*
- *Setting automatic roll over and cleanup (page 1267)*
- *Viewing and downloading log files (page 1268)*
- *Manually rolling over the log files (page 1269)*
- *Deleting server, task and FTS log files (page 1269)*



Changes to settings will only take effect after restarting all tasks.

Setting logging levels for the Server and File Transfer Service

You set the logging level for the Replicate Serverlogs and File Transfer Service logs in **Server** view. The level you set determines what information is written to the logs. The Server logs provide information about the Qlik Replicate Server instance you are working with as opposed to individual tasks. For information on configuring the task logs, see *Setting the task logging level (page 1234)*.

The following logging levels are available, ordered from the lowest level to the highest:

1. Errors
2. Warnings
3. Info
4. Trace
5. Verbose

The higher levels always include the messages from the lower levels. Therefore, if you select **Error**, only error messages are written to the log. However, if you select **Info**, informational messages, warnings, and error messages are included. Selecting **Verbose** writes all possible messages to the log.

You can set a global logging level for all components or you can set a separate logging level for each component.

To set the logging levels:

1. On the left side of the **Server** view, click **Logging** and then click the **Server Logging Levels** or **File Transfer Service Logging Levels** sub-tab as required. The **Component Logging Level** sliders are displayed.
2. To set a global logging level, move the top slider (the slider with the labels) to the log level you want. Note that all of the sliders for the individual modules move to the same position that you set in the main slider.
3. Make any changes to the sliders for the individual modules. This is optional. Note that if you change the main slider, all of the individual sliders are reset to the new position. If you want to maintain a different logging level for a specific module, you need to reset it.
4. Click **Save** at the top of the screen. Changes to the logging level take place immediately. There is no need to restart the Qlik Replicate Server.



*Click **Save** to save the information for all settings, not only for the logging settings. If you made changes that you do not want to keep, click **Discard Changes** to discard all changes before you make changes to any of the other settings.*

Storing trace and verbose logging in memory

When the logging level is set to "Trace" or "Verbose", you can instruct Replicate to store the logging information in memory until an error occurs. On detecting an error, Replicate will begin writing to the physical logs and continue to do so for a few minutes after the initial occurrence of the error.

If *no* error occurs before the allocated memory is used up, Replicate will empty the memory buffer and start afresh.

This option is useful for tasks that fail unpredictably and for no obvious reason. The problem with continually writing large amounts of information to the logs is twofold:

- Running in "Trace" or "Verbose" logging mode will quickly use up available disk space (unless the logging settings have been configured to prevent this).
- Continually writing large amounts of data to the logs will affect performance.

To use this option

1. Select the **Store trace/verbose logging in memory, but if an error occurs, write to the logs** check box at the top of the tab.
2. In the **Allocate memory up to (MB)** field, specify the amount of memory you want to allocate for storing logging information.

Setting automatic roll over and cleanup

In the **Log File Management** sub-tab, you can define when Replicate should roll over the server log files (repsrv and repcmd) and tasks log files (reptask_<task_name>) and when to delete old log files from the system.

Automatic rollover

You can determine when to stop logging to the current log file and begin to log to a new log file. Rolled over log files are appended with a 12-digit timestamp. The current server log file is named repserv while saved (older) server log files are named repserv_XXXXXXXXXXXX.

The current running task log file is named reptask_<task_name> while saved (older) task log files are named reptask_<task_name>_XXXXXXXXXXXX.

For both server and task saved log file names, XXXXXXXXXXXX represents a 12-digit timestamp.

- **Roll over the log if the log file is older than (days):** Select the check box and then specify the maximum number of days the current log file is allowed to exist before being rolled over. The default value is **7** days.

- **Roll over the log if the log file is larger than (MB):** Select the check box and then specify the maximum number of megabytes the current log file is allowed to reach before being rolled over.



When the **Store trace/verbose logging in memory, but if an error occurs, write to the logs** option is enabled, the actual size of the **repsrv.log** may reach the sum of the **Allocate memory up to (MB)** size and the **Roll over the log if the log file is larger than (MB)** size, before it is rolled over.

For more information on the "Store trace/verbose logging in memory" option, see *Setting logging levels for the Server and File Transfer Service* (page 1266)

The default value is **100** megabytes.



- If you edit this setting while tasks are running, the new setting will not affect the task log files until the tasks are stopped and then resumed. The server log files are not affected by this limitation.
- The scheduled process (`LogFileCleanLogs`) that checks the log file size runs every five minutes. Consequently, the actual size/age of the rolled over log file may deviate slightly from the specified value(s).
- The time of a file is determined by its "modified" time as opposed to its "created" time.

Automatic cleanup

You can determine the maximum number of days old log files (i.e. log files that have been rolled over) are retained before being deleted.

- **Delete log files that are older than (days):** Select the check box and then specify the maximum number of days to retain a saved log file. Log files that are older than the specified number of days will be automatically deleted from the system. For example, if you specify **4**, then on the fifth day, any log file older than **4** days will be deleted.

The default value is **45** days.

Viewing and downloading log files

You can view log files and download them if necessary.

To view or download the log files:

1. Select the **Server Logging Levels** or **File Transfer Service Logging Level** sub-tab as required.
2. Click the **Log Viewer** toolbar button.
The **Log Viewer** window opens.
3. Continue from step 4 in *Viewing, downloading, and manually rolling over task log files* (page 1235).

Manually rolling over the log files

You can manually roll over the Replicate Server log files if necessary. This lets you stop logging to the current log file and begin to log to a new log file. The currently open log file does not have a timestamp. The name of saved (older) log files is appended with a 12-digit timestamp.

To roll over the log files:

1. Select the **Server Logging Levels** sub-tab.
2. Click the **View Logs** toolbar button.
The **Log Viewer** window opens.
3. Select the log file without a timestamp and then click the **Roll Log File** toolbar button in the top right of the window.

Deleting server, task and FTS log files

You can manually delete task, server, and File Transfer Service log files older than the specified number of days.

To delete the log files:

1. Select the **Logging** tab.
2. Click the **Delete Logs** toolbar button.
The **Delete Logs** window opens.
3. Select which logs to delete and, for each log, optionally change the default number of days (45).
4. Click **Delete**.
Selected logs older than the specified number of days will be immediately deleted.

16.5 File Transfer Service

The Qlik File Transfer Service (FTS) is a robust and reliable file transfer engine designed to efficiently transfer files over the WAN. This can dramatically improve transfer speeds when the source endpoint and the target endpoint are located on different LANs.



Changes to settings will only take affect after restarting all tasks.

How it works

A solution using FTS consists of two Qlik Replicate Servers: A local Qlik Replicate Server installed on the source endpoint LAN and a remote Qlik Replicate Server installed on the target endpoint LAN.

A local task on the local server is defined from the source endpoint to a File Channel target. A remote task on the remote Qlik Replicate Server is defined from a File Channel source to the target endpoint.

The FTS runs on the remote Qlik Replicate Server only and transfers the File Channel files from the storage location defined in the local task to the storage location defined in the remote task.

Upon file transfer, and before *Compression (page 1270)* and *Encryption (page 1270)*, large files are split into smaller blocks which form recoverable transport units, and small files are merged into bigger blocks to be sent at the same time. The blocks are then transferred and reconstructed into File Channel files when received by the FTS server.

For information on setting up a File Channel source or target to use FTS, see *Setting advanced connection properties (page 1068)* and *Setting advanced connection properties (page 1070)* respectively.

Compression

File Channel files are compressed upon sending using GZIP. You can disable the compression and control the compression level.

Encryption

After compression, File Channel files are encrypted using a randomly generated AES-256 session key. The session key is exchanged between the client and server using the Diffie-Hellman key exchange protocol which is authenticated using a secret key that is shared between the client and the server.



The File Transfer Service should be configured on the remote Qlik Replicate Server only.

Defining a File Transfer Service

Define a File Transfer Service as described below.

To add a File Transfer Service:

1. Switch to **Server** view as described in *Server view (page 96)*.
2. In the left side of the **Server** view, click **File Transfer Service**.
The File Transfer Service list is displayed.
3. In the Actions toolbar, click **Add File Transfer Service**.
The **Add File Transfer Service** window opens.
4. Edit the values in the **Name**, **Host** and **Port** columns as follows:
 - **Name:** The name of the File Transfer Service.
 - **Host:** The host name or IP address of machine on which the remote Qlik Replicate Server is installed. The default is 0.0.0.0 (all interfaces). If the server has multiple NICs (Network Interface Cards), you can define a different File Transfer Service for each card.
 - **Port:** The port through which the File Channel files are received.
 - **Enabled:** select the check box to enable the File Transfer Service.

5. Click **Save** to save your settings.

SERVER				
Save Discard Changes + Add File Transfer Service Remove				
Notifications	Name	Host	Port	Enabled
License	defaultFts	<input type="text"/>	9090	<input type="checkbox"/>
Global Error Handling				
Logging				
File Transfer Service				
File Transfer Service				
Scheduler				

Editing a File Transfer Service

You can edit a File Transfer Service as described below.

To edit a File Transfer Service:

1. Select the File Transfer Service you want to edit.
2. Edit the values in the **Name**, **Host** and **Port** columns as follows:
 - a. Click the cell to make it editable.
 - b. Change the value as required and then click **Save**.



When you edit a File Transfer Service, make sure that any File Channel targets configured to use the File Transfer Service are also updated accordingly. For more information on File Channel Targets, see [Using the File Channel as a target \(page 1069\)](#).

Deleting a File Transfer Service

You can delete File Transfer Services that you no longer want to use.

To delete a File Transfer Service:

1. In the File Transfer Services List, select the item you want to delete.
2. Click the **Remove** toolbar button.

16.6 Scheduling jobs

Use the Qlik Replicate Scheduler to schedule a one-time job or a recurrent job for specific Replicate task operations. A job is essentially an operation that can be scheduled to occur once, daily, weekly or monthly.

The following operations can be scheduled:

- Run/Resume a task
- Stop a task
- Reload a task



- Changes to settings will only take affect after restarting all tasks.
- Compose tasks are not supported.

To schedule a new job:

1. Switch to **Server** view as described in *Server view (page 96)*.
2. In the left side of the **Server** view, click the **Scheduler** tab.
The **Scheduler** tab consists of two sub-tabs: **Scheduled Jobs** and **Executed Jobs**. The **Scheduled Jobs** tab contains a list of jobs that are scheduled to run periodically or once only while the **Executed Jobs** tab contains a list of jobs that have already run.



The **Executed Jobs** tab will only show executed jobs that were scheduled to run once only. In other words, jobs scheduled to run periodically (e.g. Daily, Weekly, Monthly) will not be shown.

3. Click the **New Scheduled Job** toolbar button.
The **New Scheduled Job** window opens.
4. Specify a **Job Name** and then, from the **Select scheduled job type** drop-down list, select one of the following:

- **Run task** to run or resume the task(s) at the scheduled time.



For Full Load only tasks, it is preferable to select **Reload target** rather than **Run task** when the scheduling is set to Daily, Weekly or Monthly. This will update the table's data whereas **Run task** will replace the existing table.

- **Stop task**
- **Reload target**



Selecting **Reload target** will execute the task according to the task's replication settings. For example, if the task's **Full Load** and **Apply Changes** options are enabled, **Reload target** will reload the target tables and apply any subsequent changes.

5. Select one of the following time conventions:

- **Use server local time** - When this option is selected (the default), the job will run when the specified time is reached in the server's location.
See also: [Impact of DST Change on Qlik Replicate](#).
- **Use universal time (UTC)** - When this option is selected, the job will run at the specified UTC time. So, for example, if the server is located in a UTC + 2 timezone, a job scheduled to run at 13:00 UTC time will actually run at 15:00 local server time. Scheduling a job to run in UTC mode may be useful if you need tasks on several Replicate servers (located in different timezones) to run concurrently.



*For reference, both the server local time and the UTC time are displayed to the right of the **Scheduled Time** heading.*

6. Select and define one of following scheduling options:

- Once (Run the job once on the specified day and at the specified time)
- Daily - (Run the job every day at the specified time)
- Weekly - (Run the job on the specified days and at the specified time)
- Monthly - (Run the job on the specified day of the month)



*To run the job on the last day of every month, select **Last day of every month** from the **Day of month** drop-down list.*

7. For the **Apply to tasks** option, select which tasks to schedule. Select either **All tasks** to apply the job to all current and future tasks or **Selected tasks** to apply the job to specific tasks. If you choose **Selected tasks**, a list of currently defined tasks is displayed. Select which tasks to apply the job to.

8. Click **OK** to save your settings.

To enable or disable a scheduled job:

- In the **Scheduled Jobs** tab, select or clear the check box in the **Enabled** column as required.

To edit a scheduled job:

1. Select the job in the **Scheduled Jobs** or **Executed Jobs** list.
2. Click the **Open** toolbar button and edit the job as required.

To delete a scheduled job:

1. Select the job in the **Scheduled Jobs** or **Executed Jobs** list.
2. Click the **Delete** toolbar button.

16.7 User permissions

You can grant Qlik Replicate users different permissions according to the tasks you want them to perform. Four predefined "roles" are available: Admin, Designer, Operator and Viewer. Each role has its own set of permissions, which are described in the following table.

Default user permissions according to roles

Permissions	Admin	Designer	Operator	Viewer
View task history	Yes	Yes	Yes	Yes
Download a memory report	Yes	Yes	Yes	No
Download a Diagnostics Package	Yes	Yes	Yes	No
View and download log files	Yes	Yes	Yes	No
Perform runtime operations (such as start, stop, or reload targets)	Yes	Yes	Yes	No
Create and design tasks	Yes	Yes	No	No
Edit task description in Monitor View	Yes	Yes	No	No
Delete tasks	Yes	Yes	No	No
Export tasks	Yes	Yes	Yes	No
Import tasks	Yes	No	No	No
Change logging level	Yes	Yes	Yes	No
Delete logs	Yes	Yes	Yes	No
Manage endpoint connections (add, edit, duplicate, and delete)	Yes	Yes	No	No
Open the Manage Endpoint Connections window and view the following endpoint settings: Name, type, description, and role.	Yes	Yes	Yes	Yes
Click the Test Connection button in the Manage Endpoint Connections window.	Yes	Yes	Yes	No
View all of the endpoint settings in the Manage Endpoint Connections window.	Yes	Yes	Yes	No
Edit the following server settings: Notifications, scheduled jobs, and executed jobs.	Yes	Yes	Yes	No
Edit the following server settings: Mail server settings, default notification recipients, license registration, global error handling, log management, file transfer service, user permissions, and resource control.	Yes	No	No	No

Permissions	Admin	Designer	Operator	Viewer
Specify credentials for running operating system level post-commands on Replicate Server. For details, see <i>External utilities (page 1277)</i> .	Yes	No	No	No



The user under whose account Qlik Replicate is installed will be associated with the Admin role by default.

You can set user permissions using Active Directory Groups or Local Groups. To set user permissions using Active Directory groups, you can either create Active Directory groups with the names listed in the following table or you can create Active Directory groups with different names. Then, add users to the groups according to the role you want them to perform.

Roles and Active Directory Groups

Role	Active Directory Groups
Administrator	AttunityReplicateAdmins
Designer	AttunityReplicateDesigners
Operator	AttunityReplicateOperators
Viewer	AttunityReplicateViewers

For information on encrypting user permissions, see *Encrypting the user permissions file (page 74)*.



If you create Active Directory groups with different names, you need to add them to the **User Permissions** window and set their permissions as described in *Managing user permissions (page 1275)*.

Managing user permissions

This section explains how to edit user permissions as well as how to add or remove users and groups.

To edit the user permissions:

1. Switch to **Server** view as described in *Server view (page 96)*.
2. In the left side of the **Server** view, click the **User Permissions** tab.
3. Adjust the permission sliders as desired.
4. Click **Save** to save your settings or **Discard** to revert them.

To add new users or groups:

1. Switch to **Server** view as described in *Server view (page 96)*.
2. In the left side of the **Server** view, click the **User Permissions** tab.

3. Click the **Add toolbar** button.
The **Add User/Group** dialog box opens.
4. Select **User** or **Group** as appropriate.
5. Enter the user or group name in the following format:
For domain users/groups: domain\group_name or domain\user_name



Active Directory distribution groups are not supported.

For local users/groups: computer_name\group_name or computer_name\user_name

Then click **OK**.

The user/group is added to the **User/Group** list.

6. Click **Save** to save your settings or **Discard** to revert them.

To remove a user or group:

1. Switch to **Server** view as described in *Server view (page 96)*.
2. In the left side of the Server view, click the **User Permissions** tab.
3. Select the user/group you want to remove and then click the **Delete** toolbar button.
The user/group is deleted.
4. Click **Save** to save your settings or **Discard** to revert them.

16.8 Resource control

You can set high and critical disk space and memory utilization thresholds.

Note Disk space is checked only for the drive where the data folder resides.

Thresholds are calculated as a percentage of total capacity. So, for example, a disk space utilization threshold of 80% would mean that 20% of available disk space remains.

After setting the thresholds, you can click the **New Notification** button to [define a notification](#) that will be sent whenever a given threshold is exceeded and/or returns to normal.



Changes to settings will only take effect after restarting all tasks.

Disk space

In the **High Disk Space Utilization Threshold** section, specify the high disk space utilization threshold (in terms of percentage). When the threshold is reached, a notification will be sent (if defined).

In the **Critical Disk Space Utilization Threshold** section, specify the critical disk space utilization threshold (in terms of percentage). When the threshold is reached, all tasks will be stopped and a notification will be sent (if enabled). Replicate will resume the tasks automatically when there is sufficient disk space to do so.

System memory

Memory utilization is calculated using the following formula (note that “swap file” is used generically to refer to both page file memory on Windows and swap file memory on Linux):

$$\frac{(\text{used_swap_file} + \text{used_physical_memory})}{(\text{total_swap_file} + \text{total_physical_memory})} * 100$$

Example:

$$(5 \text{ GB} + 5 \text{ GB}) / (10 \text{ GB} + 10 \text{ GB}) * 100 = 50\%$$

In the **High System Memory Utilization Threshold** section, specify the high system memory utilization threshold (in terms of percentage). When the threshold is reached, a notification will be sent (if defined).

In the **Critical System Memory Utilization Threshold** section, specify the critical system memory utilization threshold (in terms of percentage). When the threshold is reached, Replicate will start stopping tasks and a notification will be sent (if enabled). The tasks will be resumed automatically when there is sufficient memory to do so.

16.9 External utilities

Several endpoints support running user-defined commands for pre/post-processing files. In this tab, you can provide user credentials for user-defined commands that needs to be executed at operating system level.

By default, such commands are executed under the Replicate Server service account. This may constitute a security risk, as it allows any Replicate user with Admin or Designer permissions to specify user-defined commands that could lead to a full compromise of the server.

Note that when Replicate Server is installed on Linux, this tab will be hidden as the potential security risk is relevant to Windows only.



Changes to settings will only take affect after restarting all tasks.

Endpoints that support user-defined commands are as follows:

- Amazon S3 target
- File source
- File target
- Google Cloud Storage target
- Microsoft Azure ADLS target



IMPORTANT Execution of user-defined commands is blocked by default. To allow execution of user-defined commands, follow the procedure described in *Controlling execution of user-defined commands* (page 79).

To run user-defined commands under the Replicate service (Log On) account (the default):

- Select **Replicate service account**.

To run user-defined commands under a different account:

1. Select **Another account** and then specify a user name and password in the designated fields. For domain users, the user name should be entered in the following format: domain/Username. For local users, the domain is not required.
2. If you [changed the default Attunity Replicate Server service Log On user](#), add the user specified in the Attunity Replicate Server service **Log On** tab to the "Replace a process level token" policy in the "Local Security Policy" settings.

16.10 Endpoints

The Endpoints tab lets you configure settings related to endpoint such as Endpoint Servers and default proxy settings.

Endpoint Servers

An Endpoint Server is a Java service that is installed with Replicate and starts automatically if an appropriate license is detected. The Endpoint Server is responsible for managing a set of custom endpoint types (developed using the Qlik Replicate Custom Endpoint SDK).



- *The procedure for adding custom endpoint connections and configuring tasks is the same as that for adding regular Replicate endpoints/tasks.*
- *In future versions, the **Add Endpoint Server**, **Edit**, and **Remove** toolbar options will allow you to add and manage additional endpoint servers. In this version however, they should only be used if requested by a Qlik Support Engineer.*
- *Changes to settings will only take affect after restarting all tasks.*

Viewing Endpoint Server properties

When the Endpoint Server is selected, the right pane will display all associated endpoint types. Additionally, the following properties will be displayed in the grid:

- **Status** – Indicates whether the Endpoint Server is running, stopped or in error.
- **Name** – The name of the Endpoint Server.
- **Host** – The host machine on which the Endpoint Server is installed.
- **Port** – The port of the Endpoint Server.

- **Version** – The version of the Endpoint Server.
- **Message** – Displayed if the Endpoint Server is in an error state or Replicate cannot connect to it.
- **Last Connection** – The last time Replicate successfully connected to the Endpoint Server.

JVM configuration options

The AT_JVM_OPT system environment variable enables you to set additional JVM configuration options such as limiting the heap memory consumed by the endpoint server, setting verbose logging, and so on. AT_JVM_OPT is supported on both Windows and Linux and is set on the former using the "set" command, and on the latter using the "export" command.

For example, to set AT_JVM_OPT with verbose logging on Windows:

1. Issue the following command:
`set AT_JVM_OPT=-verbose:cl`
2. Restart the Qlik Replicate Server service.

Endpoint Server logging

The following topic explains how to set Endpoint Server logging levels as well as how to manage and view associated log files.

Before you can perform the actions described in this topic, first switch to **Server** view (as described in *Server view (page 96)*) and then select the **Endpoint Servers** tab.

Setting logging levels

To set the logging levels for an Endpoint Server:

1. Select the desired Endpoint Server from the grid.
2. In the Actions toolbar, click **Logging Settings**.
3. Click the **Logging Levels** tab.
4. Set the logging level for an individual component by moving the slider to the desired level next to the component name.
OR
Set a global logging level for all components by moving the top-most slider to the desired level.
5. Click **Save**.



This will set the logging level for the Endpoint Server only. To modify the logging levels for a task associated with a custom endpoint type, open the Replicate task and navigate to Task Settings | [Logging](#).

Log file management

To set the rollover and retention policies for Endpoint Server log files:

1. Select the desired Endpoint Server from the grid.
2. From the Actions toolbar, click **Logging Settings**.
3. Click the **Log File Management** tab.
4. Select the **Automatic Rollover** check box to enable automatic rollover of log files and enter the maximum size for each log (in MB). To disable automatic rollover of log files, clear the check box.
5. Select the **Automatic Cleanup** check box to enable automatic cleanup of log files and enter the maximum number of log files to retain. To disable automatic cleanup of log files, clear the check box.
6. Click **Save**.

Viewing log files

To view the log files for an Endpoint Server:

1. Select the desired Endpoint Server from the grid.
2. From the Actions toolbar, click **View Logs**.
3. Select the log file you want to view or download from the list in the **Log Files** pane. The contents of the log file will be displayed in the right pane.
4. Browse through the log file using the scroll bar on the right and the navigation buttons at the top of the window.
5. To search for specific text in the log file, enter the text in the search box; any matching text will be highlighted blue.
6. To download the log file, click the **Download Log File** icon in the top right of the window.
7. To manually roll over the current log file, click the **Roll Log File** icon in the top right of the window.



Selecting **View Logs** will display Endpoint Server logs only.

To view log files for individual tasks on the Endpoint Server machine:

1. Open the task.
2. Switch to Monitor view.
3. Click **View Logs**.
4. Select **Source Endpoint Server** from the drop-down list.

Default proxy server

You can decide whether to configure proxy settings for each endpoint or whether to use the setting configured in this tab.

Supported endpoints

Supported with the following endpoints only:

- Microsoft Azure Synapse Analytics target
- Google Cloud BigQuery target
- Databricks Lakehouse (Delta) target
- Databricks (Cloud Storage) target
- Amazon Redshift target

Settings

The following settings are available:

- **Host name:** The host name of the proxy server.
- **Port:** The port via which to access the proxy server.
- **User name:** The user name for the proxy server.
- **Password:** The password for the proxy server.
- **Scheme:** Select which protocol to use to access the proxy server (HTTP or HTTPS).



If the endpoint does not support HTTPS, HTTP will be used instead.

16.11 Transformations and Filters

Click the **Global Task Settings** tab to show the **Transformations and Filters** tab. Expressions can be used in global transformations and filter rules as well as in table-level transformations and filters.

- **Support special characters in column names used in expressions:** Enable this option if you want to include source column names with special characters (for example, `special#column`) in expressions. You can override this setting for specific tasks in the *Transformations and Filters* (page 1201) tab.



This option requires that all table and global transformations must use strict SQL syntax. Specifically, all string literals must use single quotes (') rather than double quotes. If any of the transformations on any task on the server uses double quotes (") to quote literals, then you should either not enable this option or fix the double-quotes as described below.

Example of replacing double quotes with single quotes in an expression:

This expression

```
CASE age WHEN age < 18 THEN "adult" ELSE "minor" END)
```

must be written as:

```
CASE age WHEN age < 18 THEN 'adult' ELSE 'minor' END)
```

See also:

- *Defining global rules (page 1107)*
- *Using filters (page 1092)*
- *Defining transformations for a single table/view (page 1084)*
- *Using the Expression Builder (page 1137)*

16.12 More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

More options

These options are not exposed in the UI as they are only relevant to specific versions or environments. Consequently, do not set these options unless explicitly instructed to do so by Qlik Support or product documentation.

To set an option, simply copy the option into the **Add feature name** field and click **Add**. Then set the value or enable the option according to the instructions you received.

A Using Change Tables

You can use Qlik Replicate tasks to save the change events in change tables. This section describes how to use Qlik Replicate with change tables.

In this section:

- *Working with Change Tables (page 1283)*
- *Change Tables (page 1284)*
- *Usage example (page 1287)*

A.1 Working with Change Tables

In addition to replicating changes from source endpoint tables to corresponding tables in a target endpoint, Qlik Replicate can also replicate changes to corresponding Change Tables in the target endpoint. This process occurs simultaneously when applying changes to the target tables. Qlik Replicate lets you determine whether to replicate the changes to the target only, store changes in the Change Tables, or both. See *Using the Change Table model (page 1284)* below for more information.



UPDATES applied to the source that do not change the source data will be applied to the target but will not be applied to the corresponding Change Table. For example, if an UPDATE operation on Column A in the source changes all values greater than 10 to 1, and one of the records in Column A is already 1, then the UPDATE for that record will not be written to the Change Table.

The Change Tables have the same names as the tables that are being replicated, however a suffix is added to each table name. By default the suffix is `__ct`, however you can change the suffix name in the Qlik Replicate Console. For more information on changing the suffix added to the Change Table names, see *Store Changes Settings (page 1175)*.

In addition to the selected columns from the source table, the Change Table also includes special header columns that provide more information on the change the row represents such as the operation, the transaction and the timestamp. This lets you use SQL Query Language to carry out various analysis of the change events, such as fraud detection, trend analysis, triggering of business processes, and Disaster Recovery. For more information about reading the Change Tables, see *Change Tables (page 1284)*.

Handling truncate operations

TRUNCATE operations will *not* truncate the Change Table. Instead, an additional record will be added to the table with `operation=TRUNCATE`. This also means that, when replicating to a Hadoop target, the HDFS files corresponding to the Change Table will *not* be deleted.

Regarding the actual target table, if both the Apply Changes *and* Store Changes replication options are enabled, the target table will be truncated. This also means that, when replicating to a Hadoop target, the HDFS files corresponding to the Change Table will also be deleted.

To apply TRUNCATE operations to both the Change Table and the Target Table (for sources that support TRUNCATE):

1. In the task settings' [Store Changes Settings tab](#), make sure that **Apply to Change Table** (the default) is selected from the **DDL options** drop-down list.
2. In the task settings' [Apply Changes Settings tab](#), make sure that **TRUNCATE target table** (the default) is selected from the **When source table is truncated** drop-down list.

Using the Change Table model

When you work with Change Tables you can determine whether to store the changes in the Change Tables, apply the changes to the target tables, or both store and apply the changes. You determine this when you define the replication task. For more information on this setting, see *Store Changes Settings (page 1175)*.

In cases where you are both applying and storing the changes, the following is true:

- The target and Change Tables must be in the same endpoint, although they can have different schemas. For example, the Change Tables will contain the metadata headers. For further details about changing the schema, see *Rename Change Table schema (page 1129)*.
- Changes applied to the Change Table will be handled exactly the same as the changes performed in the corresponding transaction in the source database. Therefore, when using **Transactional apply mode** or **Batch optimized apply** mode with the **Preserve transaction consistency** option selected, the changes will be processed as a single transaction. The exception to this is when an error is encountered and Replicate switches to "one-by-one" apply mode in order to determine which of the Change operations is responsible for the error.
- The same data columns are both applied and stored with the exception of the change header columns, which are only added to the stored Change Tables.


A.2 Change Tables

You can use the tools for your target endpoint to get information using the metadata in the change tables. This data is defined by the header columns added to the change table schema. These headers are described in the table below.

For every target table in the replication task, a change table with the corresponding name is maintained by Qlik Replicate in the endpoint with the target tables. For more information, see *Working with Change Tables (page 1283)*. A change table contains the original table columns, and header columns. The header columns contain a prefix so that the name does not conflict with the source table column names. The default prefix is `header__`. For information on how to change this prefix, see the **Change tables** listing under *Metadata (page 1166)* in *Task Settings (page 1165)*. The following table lists the default change table header columns.

Column Name	Type	Description
[header_ _]change_seq	varchar (35)	<p>A monotonically increasing change sequencer that is common to all change tables of a task. The change sequence has the following format:</p> <p>YYYYMMDDHHmmSShhxxxxxxxxxxxxxxxxxxxx</p> <p>Where:</p> <ul style="list-style-type: none"> • YYYY is the four-digit year (such as 2012) • MM is the two-digit month (range from 01-12) • DD is the two-digit day (range from 01-31) • HH is the hour in the day (range from 00-23) • mm is the minute in the hour (range from 00-59) • SS is the second in the minute (range from 00-59) • hh is the hundredth of the second (range from 00-99) • xxxxxxxxxxxxxxxxxxx is a 19-digit, zero prefixed change number (global per task) <p>The time part usually refers to the commit time of the transaction that includes the change record. Replicate contains logic that maintains the monotonicity of the sequence number so modifying or adjusting the endpoint time may result in multiple changes to seem that they are within the same timestamp but with increasing change number.</p> <p>The xxx...xxx is usually the internal change number from the data record except that for BEFORE-IMAGE records it is the same as the change number of the matching UPDATE record (for example, if the change number of BEFORE-IMAGE is 1000 and that of the UPDATE is 1001, then both have 1001). This allows a simple left-outer-join between the table and itself where on the left we scan until the point in time but filter out <code>operation=before-image</code>, and on the right we join on the same <code>change_seq</code> with the <code>change_oper</code> being 'B' .</p>
[header_ _]change_oper	varchar (1)	<p>The operation type. This can be one of the following:</p> <ul style="list-style-type: none"> • I: INSERT • D: DELETE • U: UPDATE • B: Before Image

Column Name	Type	Description
[header__] change_mask	varbinary (128)	<p>The change mask indicates which data columns in the change table are associated with columns that changed in the source table.</p> <p>The bit position in the change mask is based on the column ordinal in the change table. This means that if there are 5 header columns, they occupy bits 0 to 4 and the first data column is bit 5 in the change mask.</p> <p>The change mask is a binary column (a byte array) representing the change mask in little-endian order:</p> <p>Byte 0 bit7 bit6 bit5 bit4 bit3 bit2 bit1 bit0</p> <p>Byte 1 bit15 bit14 bit13 bit12 bit11 bit10 bit9 bit8</p> <p>In this example, <code>bit#N</code> indicates that the change table column of ordinal N relates to a column that changed in the source table. If update mask is 11000 and the column ordinal is 3 the column did not change.</p> <p>The following describes the bit semantics:</p> <ul style="list-style-type: none"> • For INSERT records, all the inserted columns have the associated bits set. • For DELETE records only primary-key (or unique index) columns have the associated bits set. This allows an applier to construct a DELETE statement without having to find the primary key fields from another source. • For BEFORE-IMAGE records, all bits are clear (the change mask can be empty). • For UPDATE records, each column whose value changed between the BEFORE-IMAGE and the UPDATE will have the associated bit set. <p>For space and processing efficiency, the actual number of bytes stored in the change-mask can be null-trimmed. This means that trailing zeros do not need to be stored. The handling logic should take this into consideration.</p>
[header__] stream_ position	varchar (128)	The source CDC stream position.

Column Name	Type	Description
[header__] operation	varchar (12)	The operation associated with the change record. It can be one of the following: <ul style="list-style-type: none"> • INSERT • UPDATE • DELETE • BEFOREIMAGE
[header__] transaction_ id	varchar (32)	The ID of the transaction that the change record belongs to. The value is a hex-string of the 128-bit transaction ID.
[header__] timestamp	timestamp	The original change UTC timestamp (the value may be approximate). <div style="border: 1px solid #ccc; padding: 5px; margin-top: 10px;">  <i>With PostgreSQL source, the timestamp is only known after the commit occurs. Therefore, until the changes are committed to the source tables, the default date will be displayed (e.g. 1970-01-01). For subsequent transaction events, the previous commit timestamp will be shown.</i> </div>
[header__] partition_ name	string	The name of the partition created on the target when Change Data Partitioning is enabled. The partition name consists of the partition start and end time. Example: 20170313T123000_20170313T170000

A.3 Usage example

The following SQL statement returns a range of changes that starts after the last handled event (of change sequence "2012072314452211000000000000901203") and until the event committed on 23-Jul-2012 at 23:00:00.00. For update we also get the before image value (here for the salary before and after).

```
SELECT CHANGE.[header__change_seq],
CHANGE.[header__stream_position],
CHANGE.[header__operation],
CHANGE.[header__transaction_id],
CHANGE.[header__timestamp],
CHANGE.[EMPLOYEE_ID],
CHANGE.[FIRST_NAME],
CHANGE.[LAST_NAME],
```

```
CHANGE.[SALARY],
BI.[SALARY],
FROM [Replication].[HR].[EMPLOYEES_ct] CHANGE LEFT OUTER JOIN
[Replication].[HR].[EMPLOYEES_ct] BI ON
BI.[header__change_seq] = CHANGE.[header$__change_seq] AND
BI.[header__change_oper] = 'B'
WHERE CHANGE.header__oper <> 'B' AND
CHANGE.[header__stream_position] > '20120723144522110000000000000901203' AND
CHANGE.[header__stream_position] <= '2012072323000000Z' AND
ORDER BY
CHANGE.[header__stream_position], CHANGE.[header$__stream_oper]
```



B Using an Audit table

When defining a replication task, you can choose to store changes to all tables in a single Audit table (located in the target endpoint). The changes can then be pushed from a single stream to any queue (JMS for example).

The following table describes the structure of the Audit table.

Audit table structure

Column Name	Type	Description
task_name	varchar (128)	The name of the replication task.
change_seq	bigint	<p>A monotonically increasing change sequencer that is common to all change tables of a task. The change sequence has the following format:</p> <p>YYYYMMDDHHmmSShhxxxxxxxxxxxxxxxxxxxx</p> <p>Where:</p> <ul style="list-style-type: none"> • YYYY is the four-digit year (such as 2012) • MM is the two-digit month (range from 01-12) • DD is the two-digit day (range from 01-31) • HH is the hour in the day (range from 00-23) • mm is the minute in the hour (range from 00-59) • SS is the second in the minute (range from 00-59) • hh is the hundredth of the second (range from 00-99) • xxxxxxxxxxxxxxxxxxx is a 19-digit, zero prefixed change number (global per task) <p>The time part usually refers to the commit time of the transaction that includes the change record. Replicate contains logic that maintains the monotonicity of the sequence number so modifying or adjusting the endpoint time may result in multiple changes to seem that they are within the same timestamp but with increasing change number.</p> <p>The xxx...xxx is usually the internal change number from the data record except that for BEFORE-IMAGE records it is the same as the change number of the matching UPDATE record (for example, if the change number of BEFORE-IMAGE is 1000 and that of the UPDATE is 1001, then both have 1001). This allows a simple left-outer-join between the table and itself where on the left we scan until the point in time but filter out <code>operation=before-image</code>, and on the right we join on the same <code>change_seq</code> with the <code>change_oper</code> being 'B'.</p>

Column Name	Type	Description
change_oper	varchar (1)	The operation type. This can be one of the following: I: INSERT D: DELETE U: UPDATE B: Before Image
stream_position	varchar (128)	The source CDC stream position.
schema_name	nvarchar (128)	The name of the source table schema.
table_name	nvarchar	The name of the source table.
operation	varchar (12)	The operation associated with the change record. It can be one of the following: <ul style="list-style-type: none"> • INSERT • UPDATE • DELETE • BEFOREIMAGE
transaction_id	varchar (32)	The ID of the transaction that the change record belongs to. The value is a hex-string of the 128-bit transaction ID.
timestamp	timestamp	The original change timestamp (the value may be approximate).
change_record	nclob	The new data. <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;">  <i>LOB columns with unlimited size are not supported in the change_record field. The other fields will be recorded but the LOB will have a NULL value.</i> </div>
bu_change_record	nclob	The data before the update. <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;">  <i>LOB columns with unlimited size are not supported in the bu_change_record field. The other fields will be recorded but the LOB will have a NULL value.</i> </div>
Partition start		Partition name, partition start time (UTC), and estimated partition end time (UTC)

B Using an Audit table

Column Name	Type	Description
Partition end		Partition name, partition start time (UTC), partition end time (UTC), and a list of changed tables (identified by table name and schema name).

For more information on storing changes in an audit table, see *Store Changes Settings* (page 1175).

C Creating dump files

Dump files are created when a component crashes and creates an exception code. They are useful for troubleshooting such crashes as they states at exactly what line or area of memory the crash occurred.

To create a dump file:

1. Create a new environment variable with the following name and value:
Variable: `AREP_CRASH_DUMP_TYPE`
Possible values: `FULL`, `MINI` or `NONE`
2. Restart both Replicate services.
3. This step is optional. If `AREP_CRASH_DUMP_TYPE` is set to `FULL` or `MINI`, navigate to the following registry key:
`HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\Windows Error Reporting\LocalDumps\repctl.exe`
Verify that the data value for the **DumpType** Binary Value is either 1 (`MINI`) or 2 (`FULL`).
4. After a crash occurs the dump files will be created in the following directory:
`<Product_Dir>\minidumps`



When `AREP_CRASH_DUMP_TYPE` is set to `NONE`, no dump files will be created in the event of a crash.

D Setting up Qlik Replicate in a cluster environment

This section describes how to set up Qlik Replicate in Windows and Linux clustering environments.



When building failover cluster solutions with Replicate using Windows Server Failover Cluster (WSFC) or a Linux failover cluster software, Qlik recommends using a block device (physical, virtual or iSCSI-based) for the shared Replicate DATA folder. Using NFS or SMB-based storage is not supported due to the associated latency which could greatly degrade the data transfer performance, as well as due to reduced reliability and compatibility issues. When building a cloud-based high availability solution that needs to span different availability zones, it is recommended to use a Storage-as-a-Service solution that can handle the block-level replication of the storage and that is integrated with the chosen failover clustering software.

In this section:

- *Setting up Qlik Replicate in a Windows Server cluster (HA) (page 1293)*
- *Setting up Qlik Replicate in a Linux cluster (page 1297)*

D.1 Setting up Qlik Replicate in a Windows Server cluster (HA)

This section describes how to set up Qlik Replicate in a Windows Server Cluster cluster environment. For instructions on setting up a Windows clustering environment, refer to the Microsoft Help.

The steps should be performed in the order they appear below:

- *Step 1: Install Qlik Replicate in the cluster (page 1293)*
- *Step 2: Add the Qlik Replicate services (page 1295)*
- *Step 3: Define the dependencies for each service (page 1295)*
- *Step 4: Enable different console addresses in a high availability environment (page 1296)*

Step 1: Install Qlik Replicate in the cluster

To install Qlik Replicate in your cluster environment, perform steps 1-3 on each of the cluster nodes:

1. Run the Qlik Replicate setup wizard on the first node that is part of your cluster (the active node). Install Qlik Replicate to a local folder. The default path is `C:\Program Files\Attunity\Replicate`. In the setup wizard, specify a shared storage location for the "data" folder; for example, `F:\Replicate\data`. For more information, see *Setting up Qlik Replicate on Windows (page 40)*.
2. Once setup has completed, open a command prompt as an administrator and change the working directory to the Replicate **bin** directory. The default path is `C:\Program Files\Attunity\Replicate\bin`. Then, run the following three commands, where **Command 1** should only be run on the *first* cluster node.

D Setting up Qlik Replicate in a cluster environment



- The `new_master_user_key` must be identical on all participating cluster nodes. For more information on the role played by the master user key in Replicate, see *Encrypting the user permissions file* (page 74).
- Setting the admin password using the `setserverpassword` command will invalidate the master key. Therefore, make sure to set the admin password before running the `setmasterkey` command.
- The `setmasterkey` command overrides the **mk.dat** file, rendering all stored secrets invalid (as they were encrypted using the old key). Therefore, after changing the master key, you need to reenter the passwords in all of the relevant places. For more information, see *Changing and protecting the master key* (page 72). Additionally, because the Replicate server password is also stored in the **mk.dat** file, you need to reset it as described in *Changing the server password* (page 69)

Command 1:

```
repctl -d replicate_data_folder setmasterkey new_master_key master_key_scope=1
```

Where:

- `replicate_data_folder` is the shared storage location of the Replicate **data** folder defined earlier (for example, F:\Replicate\data). Paths with spaces must be enclosed with quotation marks.
- `new_master_key` is the new master key

Command 2:

```
repuctl -d replicate_data_folder masterkey set -p new_master_user_key
```

Where:

- `replicate_data_folder` is the shared storage location of the Replicate **data** folder defined earlier (for example, F:\Replicate\data). Paths with spaces must be enclosed with quotation marks.
- `new_master_user_key` is the new master user key

Command 3:



This command is only required if you are using one of the following source endpoints: MongoDB, Salesforce, or Sap Extractor. Before running this command, make sure you have registered your license.

Change your working directory to `<REPLICATE_INSTALL>\endpoint_srv\bin` and then run the following command to add the Endpoint Server **data** folder:

```
repndctl -d replicate_endpoint_server_folder masterkey set new_master_key
```

Where:

`replicate_endpoint_server_folder` is the path to the **endpoint_srv** folder (as shown in the example below):

D Setting up Qlik Replicate in a cluster environment

Example:

```
repndctl -d F:\Replicate\data\endpoint_srv masterkey set  
J8kaFbTui22Mu5vSHdEBA5fHm36h8sqD
```

3. Leave the command prompt open.
4. Stop the Replicate services.
5. Move your cluster to the next cluster node and install Qlik Replicate as described in Steps 1-2 above. Make sure you only run **Command 2** and **Command 3** (only required with certain endpoints - see [note above](#)) in Step 2.

6. After Replicate is installed on all nodes, move all resources back to node 1, where the command prompt is still open. Then run the following command (from <REPLICATE_INSTALL>\bin) to set the Client Access Point name:

```
repuictl -d replicate_data_folder configuration set --address client-access-point-name
```

Where:

- `replicate_data_folder` is the shared storage location of the Replicate **data** folder defined earlier (for example, F:\Replicate\data). Paths with spaces must be enclosed with quotation marks.
- `client-access-point-name` is the Client Access Point name for the cluster.

Step 2: Add the Qlik Replicate services

The Qlik Replicate services are called **Qlik Replicate UI Server** and **Qlik Replicate Server**.

To add the Qlik Replicate services:

1. In the left pane of the Failover Cluster Manager, select **Roles**. The available roles will be listed in the right pane of the console (for example, **QlikReplicateRoles**). Right-click the role you are working with and point to **Add a resource**. Then select **Generic Service**.
2. In the **Select Service** screen of the New Resource wizard, select **Qlik Replicate UI Server** from the List.
3. Click **Next** and follow the directions in the wizard to create the resource. For information on how to use this wizard, see the Microsoft online help.



To open Qlik Replicate Console, it is recommended to use an address that includes the name or IP address of the cluster machine (as opposed to the specific node name).

Example: `https://cluster_name_ip/attunityreplicate/`

4. Repeat the same steps to add **Qlik Replicate Server**.



For the Qlik Replicate services to be available in the list, Qlik Replicate must be installed on the computers where you defined the service.

Step 3: Define the dependencies for each service

You should define dependencies for the Qlik Replicate services. This allows the Storage and the Network names to start before the Qlik Replicate services. If these resources do not start up before the services, Qlik Replicate will not start because it will continue to search for the data location.

D Setting up Qlik Replicate in a cluster environment

To define the dependencies:

1. In the left pane of the Failover Cluster Manager console, select **Roles**. The available roles will be listed in the right pane of the console. Select the role you are working with and then, in the bottom right pane, select the **Resource** tab. From the list of the available roles, select **Qlik Replicate UI Server**.
2. Right-click the **Qlik Replicate UI Server** role and select **Properties**. The **Qlik Replicate UI Server Properties** dialog box opens.
3. In the **Qlik Replicate UI Server Properties** dialog box, select the **Dependencies** tab.
4. Click **Insert**. A new line is added to the Resource list.
5. In the Resource column, click the arrow and select the Replicate Data storage resource from the list.
6. Click **Insert** and add the Network Name resource (its name should be the same as the cluster name).
7. Repeat the steps for the Qlik Replicate Server service.
8. Start the Services using the Failover Cluster Manager and access the console using the Network name.
9. Register the license. The license should contain all host names of the cluster.



To open Qlik Replicate Console, it is recommended to use an address that includes the name or IP address of the cluster machine (as opposed to the specific node name).

Example:

https://cluster_name_ip/attunityreplicate/

Step 4: Enable different console addresses in a high availability environment

In a High Availability environment where Qlik Replicate Console is installed on two different machines, you might want to configure a different listening address per host machine name.

To do this:

1. Open a command prompt as an administrator and change the working directory to the Replicate **bin** directory.
2. Run the following command:

```
repuictl configuration set --address replicate-main:replicate_main_
interface replicate-failover:replicate_failover_interface
```

Syntax:

```
repuictl configuration set --address primary-machine-name:primary-
machine-address failover-machine-name:failover-machine-address
```

Example:

```
repuictl configuration set --address replicate-main:198.51.100.0
replicate-failover:203.0.113.24
```

D.2 Setting up Qlik Replicate in a Linux cluster

This section provides guidelines for setting up Qlik Replicate in a Linux cluster environment. There are several commercially available clustering solutions for Linux including Veritas Cluster Server, Red Hat Cluster Suite and IBM HACMP for Linux.

When one of the available clustering solutions is already in place, Qlik Replicate can be set up like any other cluster application while adhering to the following guidelines:

- Replicate should be installed on the participating cluster nodes.
For more information, see *Setting up Qlik Replicate on Linux (page 49)*.
- Replicate only supports the failover cluster configuration (active-passive). This means that, at any given time, the Data Movement gateway service should be running on one of the cluster nodes only.
- The Qlik Replicate "data" folder should be stored in a SAN for shared access between the cluster nodes. To change the default location of the Qlik Replicate data folder, run the following command on the primary cluster node when the installation completes:

```
./repctl -d <shared_storage_path> service start
```
- Only one instance of Replicate can be active at a given data location. The cluster software should be set so that during failover, one Replicate instance is stopped and the other is started.

E Control Tables

This section describes the Qlik Replicate Control Tables that are created on the target endpoint when the corresponding table is selected in the **Control Tables** tab.



- All Control Table timestamps are in UTC format.
- Control tables are not created in Full Load only replication tasks.
- Setting Replicate to **DROP and CREATE the table** if the target table already exists will not affect Control Tables.
- Replicate automatically creates the Control tables on the target during runtime. If you prefer to manually create the tables, make sure that all column names and table names are lower case, apart from the Apply Exceptions table (`attrep_apply_exceptions`) which should be upper case.

The Control Tables describe in this section are as follows:

- *Apply exceptions (page 1298)*
- *Replication status (page 1299)*
- *Suspended Tables (page 1300)*
- *Replication history (page 1300)*
- *Change data partitions (page 1301)*
- *DDL history (page 1302)*

E.1 Apply exceptions

Change Processing errors are recorded in the `attrep_apply_exceptions` table, which is described below.

The data in this table is never deleted.


Exceptions table

Column	Type	Description
TASK_NAME	nvchar	The name of the Qlik Replicate task.
TABLE_OWNER	nvchar	The table owner.
TABLE_NAME	nvchar	The table name.
ERROR_TIME (in UTC)	timestamp	The time the exception (error) occurred.
STATEMENT	nvchar	The statement that was being executed when the error occurred.
ERROR	nvchar	The actual error message.

E.2 Replication status

The `attrep_status` table contains the current status of each replication task and the target data. Although updates for the tasks in the `attrep_status` table are generated every few seconds, Replicate will only apply the updates *after* it has applied changes to the target tables. In some cases, this may take a few minutes.

Column descriptions

Column	Type	Description
SERVER_NAME	nvarchar	The name of the machine on which Replicate is installed.
TASK_NAME	nvarchar	The name of the Replicate task.
TASK_STATUS	varchar	<p>One of the following:</p> <ul style="list-style-type: none"> • FULL LOAD • CHANGE PROCESSING <p>Task status is FULL LOAD as long as there is at least one table in full load. After <i>all</i> tables have been loaded, the task status changes to CHANGE PROCESSING.</p>
STATUS_TIME	timestamp	When the status was last updated.
PENDING_CHANGES	int	The number of change records that were not yet applied to the target.
DISK_SWAP_SIZE	int	The amount of disk space that is occupied by old or offloaded transactions.
TASK_MEMORY	int	Current memory consumption in MB.
SOURCE_CURRENT_POSITION	varchar	The POSITION in the source endpoint that Replicate is currently reading from.
SOURCE_CURRENT_TIMESTAMP	timestamp	<p>The TIMESTAMP in the source from which Replicate is currently reading.</p> <div style="border: 1px solid gray; padding: 10px; margin-top: 10px;"> <p> <i>Replicate aligns the timestamp read from the source with the timestamp on the Replicate Server machine (according to the source latency). So, for example, if the time on the source was 12:30 and the time on the Replicate Server machine is 12:32, Replicate will add two minutes to the original source time. This will be the value in the SOURCE_CURRENT_TIMESTAMP column.</i></p> </div>
SOURCE_TAIL_POSITION	varchar	The POSITION of the oldest start transaction that is still not committed. This represents the newest position that you can revert to, without losing any changes. There may, of course, be duplicates.

Column	Type	Description
SOURCE_TAIL_TIMESTAMP	timestamp	The TIMESTAMP of the oldest start transaction that is still not committed. This represents the newest TIMESTAMP that you can revert to, without losing any changes. There may, of course, be duplicates.
SOURCE_TIMESTAMP_APPLIED	timestamp	This is the timestamp of the last transaction committed. In a bulk apply this will be the timestamp for the commit of the last transaction in that batch. It will only be changed as part of the last transaction in the batch.

E.3 Suspended Tables

When a table is suspended, information about the table including the reason for its suspension is recorded in the `attrep_suspended_tables` table. If a table is suspended and then unsuspended while the task is running, the entries for that table will be deleted from the `attrep_suspended_tables` table.

When a task with suspended tables stops, records for that task will remain in the `attrep_suspended_tables` table. In the event that the tables are still suspended when the task is restarted, the data is deleted and then recorded again.

Column descriptions

Column	Type	Description
SERVER_NAME	nvarchar	The name of the machine on which Replicate is installed.
TASK_NAME	nvarchar	The name of the Replicate task.
TABLE_OWNER	nvarchar	The owner of the suspended table.
TABLE_NAME	nvarchar	The name of the suspended table.
SUSPEND_REASON	varchar	The reason why the table was suspended.
SUSPEND_TIMESTAMP	timestamp	The date and time the table was suspended.

E.4 Replication history

The `attrep_history` table provides statistics about each task, such as the number and volume of records processed during a particular timeslot.

A new record is appended to the table at the end of each `TIMESLOT_DURATION` (page 1301). In other words, the data in this table is never deleted.

Column descriptions

Column	Type	Description
SERVER_NAME	nvarchar	The name of the machine on which Replicate is installed.
TASK_NAME	nvarchar	The name of the Replicate task.

Column	Type	Description
TIMESLOT_ TYPE	varchar	One of the following: <ul style="list-style-type: none"> FULL LOAD CHANGE PROCESSING (CDC) When FULL LOAD and CHANGE PROCESSING are running in parallel (some tables in full load, some in CDC), two history records will occupy the same time slot.
TIMESLOT	timestamp	The end timestamp of the time slot.
TIMESLOT_ DURATION	int	The duration of each history record in minutes.
TIMESLOT_ LATENCY	int	The latency at the end of the time slot in seconds. This is only applicable to CDC time slots. Note that this value contains the value of the <i>target</i> latency only.
TIMESLOT_ RECORDS	int	The number of records processed during the time slot.
TIMESLOT_ VOLUME	int	The volume of data processed in MB.

E.5 Change data partitions

The **attrep_cdc_partitions** table contains records of partitions created on the target database when *Change Data Partitioning* (page 1177) is enabled for a Replicate task. You can use this information to identify partitioned data that needs to be further processed.

attrep_cdc_partitions table fields

Column	Type	Description
SERVER_NAME	STRING	The name of the machine on which Replicate is installed.
TASK_NAME	STRING	The name of the Replicate task.
PARTITION_NAME	STRING	The partition name consists of the partition start and end time. Example: 20170313T123000_20170313T170000
PARTITION_START_TIME	TIMESTAMP	When the partition was opened: Example: 2017-03-13 12:30:00.000

Column	Type	Description
PARTITION_END_TIME	TIMESTAMP	When the partition was closed: Example: 2017-03-13 17:00:00.000
TABLE_OWNER	STRING	The table schema or owner.
TABLE_NAME	STRING	The table name.

E.6 DDL history

The **attrep_ddl_history** table contains a history of DDL changes that occurred in the source during replication to the target.



The DDL History table is currently supported with the following target endpoints only:

- Amazon EMR
- Cloudera Data Platform (CDP) Private Cloud
- Confluent Cloud
- Databricks (Cloud Storage)
- Databricks Lakehouse (Delta)
- Google Cloud Pub/Sub
- Google Dataproc
- Hadoop
- Hortonworks Data Platform
- Microsoft Azure Databricks
- Microsoft Azure HDInsight
- Snowflake on Google

A new record is inserted into the table whenever a supported DDL change occurs in the source. Multiple ALTER TABLE statements that occur during a task may be represented as a single row in the control table. The JSON buffer (see below) describes all the changes that occurred (e.g. ADD COLUMN A, DROP COLUMN B, ALTER COLUMN C).

For information on enabling the DDL History Control Table as well as its limitations, see *Control tables* (page 1169).



*When the **Apply Changes** task option is enabled, an entry is created for the base table (e.g. tblT1). If the **Store Changes** task option is also enabled, an additional entry is created for the CT table (e.g. tblT1__CT).*

Column descriptions

Column	Type	Description
SERVER_NAME	STRING	The name of the machine on which Qlik Replicate is installed.
TASK_NAME	STRING	The name of the Qlik Replicate task.
TABLE_OWNER	STRING	The source table schema or owner.
TABLE_NAME	STRING	The source table name. If the table was renamed, this will be the table name before the change.
CHANGE_SEQ	STRING	See Change_Seq in <i>Using Change Tables (page 1283)</i> .
TIMESTAMP	TIMESTAMP	When the change occurred.
TABLE_VERSION	INTEGER	Replicate assigns an internal version number to the table. The version number increases whenever a DDL change occurs in the source table.
DDL_TYPE	STRING	CREATE_TABLE, DROP_TABLE, ALTER_TABLE, TRUNCATE_TABLE

Column	Type	Description
DETAILS	CLOB	<p>JSON document describing the change(s)</p> <p>Example:</p> <pre>{ "owner": "string", "table": "string", "tableNewName": "string", "version": number, "columns": { "col1": { "columnNewName": "string", "action": "string", "type": "string", "length": number, "precision": number, "scale": number, "primaryKeyPosition": number, "nullable": boolean, "ordinal": number }, "col2": { ... } } }</pre> <p>where:</p> <ul style="list-style-type: none"> • <code>tableNewName</code> is NULL if the table was not renamed • <code>col1</code> and <code>col2</code> are the original column names (or the <i>only</i> column names if the columns were not renamed) • <code>columnNewName</code> is the new column name or NULL if the column was not renamed • <code>action</code> is ADD (also for CREATE TABLE), DROP, or ALTER • <code>type</code> is the Replicate data type • <code>primaryKeyPosition</code> is ZERO if the column is not part of the primary key

F Using HP NonStop SQL/MP as an ODBC target

HP NonStop SQL/MP can be used as an ODBC target in a Replicate task. However, to ensure that Replicate and non-HP NonStop SQL/MP sources comply with HP NonStop SQL/MP conventions, several additional steps need to be performed. In addition, when defining the task settings, certain options will be unavailable or limited in their availability.

Note that when replicating to an HP NonStop SQL/MP target, you must select **SQLMP (ARC)** as the provider type. For information on how to select a provider type as well as a detailed explanation of how to define HP NonStop SQL/MP as an ODBC target in a Replicate task, see *Using ODBC to connect to a target* (page 960).

In this section:

- *Prerequisites* (page 1305)
- *Table settings* (page 1306)
- *Task setting limitations* (page 1307)

F.1 Prerequisites

Before designing a task with HP NonStop SQL/MP as an ODBC Target, the following prerequisites must be met:

- All source tables and their indexes must already exist on the target database.
- The default name for the Replicate Apply Exceptions table (`attrep_apply_exceptions`) is not compliant with HP NonStop SQL/MP table naming conventions which dictate that table names cannot exceed 8 characters or contain non-alphanumeric characters. To resolve this conflict, you need to add the following entry to your NAVMAP file or create a new NAVMAP file (The NAVMAP file is located in the subvolume where ARC is installed):

Syntax:

```
[SQLMP]
```

```
attrep_apply_exceptions= \machine_name.$volume_name.subvolume_name.atreapex
```

You also need to manually create the Apply Exceptions table on the HP NonStop SQL/MP target *before starting the task*. The table should be named `atreapex` and should contain the following columns:

Column name data types

Column name	Data type
TASK_NAME	Varchar(128)
TABLE_OWNER	Varchar(128)
TABLE_NAME	Varchar(128)
ERRORTIME	Datetime year to fraction
STATEMENT	Varchar(1000)
ERROR	Varchar(1000)

- If you intend to enable the Replicate Store Changes option, you need to manually create the audit table on the HP NonStop SQL/MP target *before starting the task*.



This also requires you make the following changes in the *Store Changes Settings* (page 1175) tab:

1. Select **Audit table** from the **Store changes** drop-down list.
2. Change the default audit table name from `attrep_audit_table` to `atreauta`.

Create a table named `atreauta` with the following parameters:

Parameter data types

Column Name	Data Type
*task_name	Varchar(110)
*stream_position	Varchar(128)
change_seq	Varchar(35)
change_oper	Varchar(1)
schema_name	Varchar(128)
table_name	Varchar(128)
operation	Varchar(12)
transaction_id	Varchar(32)
timestamp	Timestamp
change_record	Varchar(1000)
bu_change_record	Varchar(1000)



: You also need to create a Unique Index consisting of the **task_name** and **stream_position** columns (marked with an asterisk above).

F.2 Table settings

When the source database in a Replicate task is not HP NonStop SQL/MP, you must make sure that the selected source tables comply with HP NonStop SQL/MP conventions and limitations. This section describes how to apply transformations to source tables that are not HP NonStop SQL/MP-compliant.

- The maximum size of a Unique Index in HP NonStop SQL/MP cannot exceed 240 bytes. In the Transform tab of the **<Table_Name> Table Settings** dialog box, you can check which columns comprise the Unique Index and change them if necessary (i.e. if their combined **Type** value exceeds 240 bytes). For an explanation of how to do this, see *Using the Transform tab* (page 1086).
- HP NonStop SQL/MP does not support UPDATE operations on Primary Key columns. Therefore, if your application updates source tables columns that are part of a Primary Key, you will need to create the target table with a Unique Index on those columns instead of a Primary Key. For an explanation of how to do this, see *Using the Transform tab* (page 1086).

- Valid HP NonStop SQL/MP table names cannot exceed 8 characters or contain non-alphanumeric characters. If any of the source table names are not HP NonStop SQL/MP-compliant, you will need to map them to a valid name.

To do this:

1. Open the **<Table_Name> Table Settings** dialog box as described in *Table Settings (page 1082)*.
2. In the **General** tab's **Table Name** field, enter the new table name.

F.3 Task setting limitations

When defining your task settings, the following limitations apply:

- **Metadata:**
 - **Target Metadata:** As HP NonStop SQL/MP does not support LOB data types, the setting in this tab are not relevant.
 - **Control Tables:** None of the optional tables are supported. The Apply Exceptions table, which is required, should be configured as described in *Prerequisites (page 1305)* above.
- **Full Load:**
 - **Full Load Settings:** As the source tables need to be created manually on the target, the **DROP and CREATE table** option is not supported.
 - **Full Load Tuning:** No limitations.
- **Change Processing:**
 - **Apply Changes Settings:** No limitations
 - **Store Changes Settings:** Change tables are not supported. If you want Replicate to store captured changes on HP NonStop SQL/MP, choose **Audit table** from the **Store changes in** drop-down list. This also requires you to manually create the Audit Table on HP NonStop SQL/MP *before starting the task*, as described in *Prerequisites (page 1305)* above. After creating the Audit table, specify its name in the **Audit table name** field (instead of the default name).

Audit table creation options:
As the Audit Table is created manually, the **DROP and CREATE audit table** option is not supported.
 - **Change Processing Tuning:** Only "Transactional apply" **Change processing mode** is supported.
- **Error Handling:** No limitations.
- **Logging:** No limitations.

For a detailed description of Task Settings, see *Task Settings (page 1165)*.

G Impact of DST change on Qlik Replicate

This topic describes the impact of Daylight Saving Time (DST) on Qlik Replicate and provides guidelines for handling changes brought about by DST.

Tasks that move from Full Load to Change Processing when DST comes into effect may encounter data loss. However, such data loss can be prevented by adhering to the guidelines outlined in this appendix.

Additionally, the times displayed in the Replicate Console may not be synced with the server. Should you encounter any time discrepancy issues, either restart the Qlik Replicate Server service or stop and resume the tasks.

There are two types of DST changes:

- **DST On** - Occurs approximately when Summer starts (actual date is country specific). Its impact on local time is that local time is moved one hour forward (so, for example, 01:00 AM becomes 02:00 AM). This DST change does not impact Qlik Replicate because it does not result in time overlap.
- **DST Off** - Occurs approximately when Winter starts (actual date is country specific). Its impact on local time is that local time is moved back one hour (so, for example, 02:00 AM becomes 01:00 AM). This DST change results in time overlap where local time travels over the same hour twice in a row.

The comments below assume that the customer has not changed the time but rather the timezone or the DST setting. Changing the actual time (not for minor time adjustments) is a sensitive operation and is best done when Qlik Replicate is stopped.

Running Qlik Replicate tasks do not depend on the timezone or DST for correctly scanning and processing the transaction logs. Internally, Qlik Replicate timers use UTC.

Still, there are several places where DST may have an effect:

1. Timestamps in logs and audit messages are in local time. As a result, when Winter time starts, the logs will show the time going back an hour; conversely, when Summer time starts, the logs may appear to be missing one hour.
2. Scheduled jobs as well as the global and table manipulation variables `timestamp` and `commit_timestamp` use local time so these will also be affected. The impact of this depends on the manipulation done and on the intended use of the timestamp based data.



To prevent timestamp and scheduling anomalies resulting from DST starting or ending, the following best practices should be observed:

- *DST Off (summer to winter): Do not schedule a task to start from the time the clock changes until the following hour. For example, if DST ends at 02:00 am, do not schedule a task to run between 02:00 and 02:59, as the task will run twice.*
- *DST On (winter to summer): Do not schedule a task to start from the time the clock changes until the following hour. For example, if DST starts at 02:00 am, do not schedule a task to run between 02:00 and 02:59 as this hour does not exist.*
- *The initial Full Load of tables or reloading of tables should not be done during the DST change window. It is recommended to perform such operations either an hour before or an hour after DST.*

If you have existing jobs scheduled to start at the overlap time and you do not want to modify them, then you need to stop the Qlik Replicate Server. Going in to Winter time, for example, if at 02:00 AM the clock is to be set back to 01:00 AM then when the time is 00:55 AM the Qlik Replicate Server should be stopped and, after an hour and ten minutes (at 01:05 AM), should be started again.

If you forget to do this, all scheduled jobs will run an hour earlier than intended. You can rectify this by setting the desired scheduling time and then restarting the Qlik Replicate Server service.

3. Statistics shown on the console are also sensitive to local time and thus may also show confusing/inaccurate data in the overlap period (going in to Winter time) or for the skipped period (going into Summer time).
4. If the clock on Qlik Replicate Server machine is one hour behind the clock on the Qlik Replicate Console (UI) machine, the following issues are known to occur:
 - The *Applied Changes graph* (page 1228) circle graph will be updated as the changes are applied, but the information in the *Recent Activity* (page 1228) tab will not be updated.
 - Scheduled jobs will start according to the Qlik Replicate Server time (as expected), but will remain in the **Active Jobs** list after execution instead of moving to the **Expired Jobs** tab.

For more information on scheduling jobs, see *Scheduling jobs* (page 1272).

In general, it is recommended to avoid non-critical task design changes during the first overlap period (going in to Winter time) so as to prevent confusion about when the changes took place.

In addition to Qlik Replicate, other components are also involved including:

- The source endpoint system
- The target endpoint system
- The local operating system
- The task design (specifically using timestamp based variables)

Given the complexity of the topic and the involvement of many independent components and settings, Qlik generally recommends that customers first verify the impact of DST changes in their test environment.

H Metadata file description

When the [Create metadata files in the target folder](#) option in the File target endpoint or Amazon S3 target endpoint is selected, for each CSV/JSON/Parquet file Replicate creates a corresponding metadata file under the specified target folder.

The metadata file offers several benefits such as enabling custom batch processes to perform better validation, supporting deeper automation, offering lineage information and improving processing reliability.

The metadata files are described in the tables below.



All timestamps are in ISO-8601 format, for example 2016-08-02T10:05:04.802.

Task Information file

Field	Description
name	The name of the Replicate task.
sourceEndpoint	The name defined in the source endpoint settings.
sourceEndpointType	The endpoint type defined in the source endpoint settings (e.g. Oracle, MySQL, etc.).
sourceEndpointUser	The user defined in the source endpoint settings.
replicationServer	The hostname of the machine on which Qlik Replicate is installed.
operation	If a target data file has been created, this field will contain the following value: dataProduced

File Information file

Field	Description
name	The name of the data file without the extension.
extension	The extension of the data file (.csv or .json according to the selected target file format).
location	The location of the data file.
startWriteTimestamp	UTC timestamp indicating when writing to the file started.
endWriteTimestamp	UTC timestamp indicating when writing to the file ended.
firstTransactionTimestamp	UTC timestamp of the first record in the file.
lastTransactionTimestamp	UTC timestamp of the last record in the file.
content	The values can either be data (i.e. Full Load replication) or changes (i.e. Change Processing replication) according to the data in the corresponding CSV file.

Field	Description
recordCount	The number of records in the file.
errorCount	The number of data errors encountered during file creation.

Format Information file

Field	Description
format	delimited or json according to the selected target file format.
options	The options for delimited file format. These options will not be shown for json format as they are not relevant.
recordDelimiter	The delimiter used to separate records (rows) in the target files. The default is newline (\n).
fieldDelimiter	The delimiter used to separate fields (columns) in the target files. The default is a comma.
nullValue	The string used to indicate a null value in the target file.
quoteChar	The character used at the beginning and end of a column. The default is the double-quote character (").
escapeChar	<p>The character used to escape a string when both the string and the column containing the string are enclosed in double quotes. Note that the string's quotation marks will be removed unless they are escaped.</p> <p>Example (where " is the quote character and \ is the escape character):</p> <p>1955,"old, \"rare\", Chevrolet", \$1000</p>

Custom Information file

Field	Description
customInfo	<p>This section contains any custom parameters that were set using the dfmCustomProperties internal parameter.</p> <p>The dfmCustomProperties internal parameter must be specified in the following format:</p> <p>Parameter1=Value1;Parameter2=Value2;Parameter3=Value3</p> <p>Example:</p> <p>color=Blue;Size=Large;Season=Spring</p> <p>For an explanation of how to set internal parameters, see <i>Internal parameters (page 707)</i>.</p>

Data Information file

Field	Description
sourceSchema	The schema containing the source table.

Field	Description
sourceTable	The name of the source table.
targetSchema	The name of the target table schema (if the source schema name was changed). For information on changing the source schema name, see <i>Performing General tasks for a single table/view (page 1083)</i> .
targetTable	The name of the target table (if the source table name was changed). For information on changing the source table name, see <i>Performing General tasks for a single table/view (page 1083)</i> .
tableVersion	Replicate assigns an internal version number to the table. The version number increases whenever a DDL change occurs in the source table.
columns	Information about the table columns.
ordinal	The position of the column in the record (1, 2, 3, etc.).
name	The column name.
type	The column data type. See <i>File target data types (page 700)</i> or Amazon S3 Target Data Types for more information.
width	The maximum size of the data (in bytes) permitted for the column.
scale	The maximum number of digits to the right of the decimal point permitted for a number.
primaryKeyPos	The position of the column in the table's Primary Key or Unique Index. The value is zero if the column is not part of the table's Primary Key.

I Supported Platforms and Endpoints

In this section:

- *Supported platforms (page 1313)*
- *Supported source endpoints (page 1314)*
- *Supported target endpoints (page 1320)*
- *Endpoints supported in bidirectional replication (page 1326)*
- *Supported browsers (page 1327)*

In addition to providing a complete list of the source and target endpoints supported by Qlik Replicate, the Support Matrix also provides information about which platforms (Windows/Linux) are supported by Qlik Replicate.

Note that depending on your license, some of the listed endpoints may not be available in your version of Qlik Replicate.



Some of the endpoints are FIPS compliant. For a list of FIPS compliant endpoints, see FIPS compliance (page 66).



For each certified major version of a supported endpoint, Qlik performs a complete set of compatibility tests on one minor version in the same release family. Once certified, any minor versions in that family will also be supported, providing backward compatibility with the major version is preserved. For example, Oracle 19.x means that Oracle 19.0, 19.3 and 19.5 are supported as well.

If an incompatibility is discovered in a minor version, a note to this effect will be added to the Support Matrix until such time the issue is resolved.

I.1 Supported platforms

Supported Windows platforms

Qlik Replicate can be installed on any of the following Windows platforms:

- Windows Server 2012 (64-bit)
- Windows Server 2012 R2 (64-bit)
- Windows Server 2016 (64-bit)
- Windows Server 2019 (64-bit)
- Windows Server 2022 (64-bit)



Windows Server 2022 is supported from Replicate May 2022 Service Release 02 only.

Supported Linux platforms

Qlik Replicate can be installed on Red Hat Enterprise Linux 8.x (64-bit) or any corresponding and compatible Linux distribution, such as CentOS.

I.2 Supported source endpoints

This section lists the source endpoint versions supported by Replicate, according to endpoint type.

- *Cloud-based (page 1314)*
- *ARC-based (page 1316)*
- *AIS-based (page 1316)*
- *File (page 1317)*
- *Data Lake (page 1317)*
- *Relational databases (page 1317)*
- *Data warehouses (page 1318)*
- *ODBC (page 1318)*
- *NoSQL (page 1319)*
- *Other (page 1319)*

Cloud-based

Supported Cloud-based sources

Endpoint	Supported versions	Windows	Red Hat Linux
Amazon Aurora MySQL Supported via the MySQL source endpoint.	Same as the MySQL source endpoint.	✓	✓
AWS Aurora Cloud for PostgreSQL	Same as the PostgreSQL source	✓	✓
Amazon RDS for MySQL	Same as the MySQL source endpoint.	✓	✓
Amazon RDS for MariaDB Supported via the MySQL source endpoint.	Same as the MySQL source endpoint	✓	✓
Amazon RDS for PostgreSQL	Same as the PostgreSQL source endpoint.	✓	✓

I Supported Platforms and Endpoints

Endpoint	Supported versions	Windows	Red Hat Linux
Amazon RDS for SQL Server	Same as the Microsoft SQL Server source endpoint (starting from 2014)	✓	✓
Amazon RDS for Oracle Supported via the Oracle source endpoint.	Same as the Oracle source endpoint (starting from 11.2)	✓	✓
Google Cloud SQL for PostgreSQL	Same as the PostgreSQL source endpoint	✓	✓
Google Cloud AlloyDB for PostgreSQL	Same as the PostgreSQL source endpoint	✓	✓
Google Cloud SQL for MySQL	Same as the MySQL source endpoint	✓	✓
Microsoft Azure SQL Managed Instance <ul style="list-style-type: none"> Use the Microsoft Azure SQL (MS-CDC) endpoint to capture changes using MS-CDC. This is the recommended method of capturing changes. Use the Microsoft Azure SQL Managed Instance endpoint to capture changes from the logs. 	Same as the Microsoft SQL Server source endpoint.	✓	✓
Microsoft Azure SQL Database Supported via the Microsoft Azure SQL (MS-CDC) endpoint.	Same as the Microsoft SQL Server source endpoint.	✓	✓
Microsoft Azure Database for MySQL	Same as the MySQL source endpoint (starting from 8.0)	✓	✓
Microsoft Azure Database for MySQL - Flexible Server Supported via the MySQL source endpoint.	N/A	✓	✓
Microsoft Azure Database for PostgreSQL Supported via the PostgreSQL source endpoint.	Same as the PostgreSQL source endpoint	✓	✓
Microsoft Azure Database for PostgreSQL - Flexible Server Supported via the PostgreSQL source endpoint.	N/A	✓	✓

I Supported Platforms and Endpoints

Endpoint	Supported versions	Windows	Red Hat Linux
MongoDB Atlas Supported via the MongoDB (Standard) source endpoint.	Same as the MongoDB (Standard) source endpoint.	✓	✓
Oracle on Oracle Cloud Supported via the Oracle source endpoint.	Same as the Oracle source endpoint (starting from 11.2)	✓	✓
Salesforce (Streaming CDC)	N/A	✓	✓
Salesforce (Incremental Load)	N/A	✓	✓

ARC-based

Supported ARC-based sources

Endpoint	Supported Versions	Windows	Red Hat Linux
IBM IMS (ARC)	IBM z/OS: 2.3 and 2.4 IMS: 13, 14, and 15.x	✓	✓
IBM VSAM Batch (ARC)	IBM z/OS: 2.3	✓	✓

AIS-based

Supported AIS-based sources

Endpoint	Supported Versions	Windows	Red Hat Linux
HP Nonstop SQL/MP (AIS) HP Nonstop Enscribe (AIS)	Himalaya: G06.32 Itanium: H06.22/J06.14	✓	✓
OpenVMS RMS (AIS)	Alpha: 8.3 Itanium: 8.3	✓	✓

File

Supported file-based sources

Endpoint	Supported Versions	Windows	Red Hat Linux
File	N/A	✓	✓
File Channel	N/A	✓	✓


Data Lake

Supported data lake sources


Endpoint	Supported Versions	Windows	Red Hat Linux
Hadoop	Please contact your Qlik Account Manager for support details.	✓	✓

Relational databases

Supported relational database sources

Endpoint	Supported Versions	Windows	Red Hat Linux
IBM DB2 for LUW	10.5, and 11.1, and 11.5  10.5 with fix pack 5 is not supported.	✓	✓
IBM DB2 for z/OS	DB2: 12.1, and 13.1 IBM z/OS: 2.3, 2.4, and 2.5	✓	✓
IBM DB2 for iSeries	7.2, 7.3, 7.4, and 7.5	✓	✓
IBM Informix	12.1, and 14.10	✓	
Microsoft SQL Server	2014, 2016, 2017, 2019, and 2022	✓	✓
Microsoft SQL Server (MS-CDC)	2014, 2016, 2017, 2019, and 2022	✓	✓
MySQL	5.7 and 8.0	✓	✓
MariaDB	10.4 and 10.5	✓	✓
Supported via the MySQL source endpoint.			

I Supported Platforms and Endpoints

Endpoint	Supported Versions	Windows	Red Hat Linux
Percona Supported via the MySQL source endpoint.	Same as the MySQL source endpoint	X	✓
Oracle	11.x (starting from 11.2.0.4), 12.x, 18.x, 19.x, 21, and 21c <div style="border: 1px solid gray; padding: 5px; margin: 5px 0;">  <i>If you set the Oracle compatibility parameter, make sure you specify a version supported by Replicate. For example, if you are working with version 11.2.0.4, you cannot set an earlier version as the compatibility version.</i> </div>	✓	✓
PostgreSQL	11.x, 12.x, 13.x, 14.x, and 15.x	✓	✓
SAP Sybase ASE	16	✓	✓
SAP HANA	1.0 and 2.0	✓	✓

Data warehouses

Supported data warehouse sources

Endpoint	Supported Versions	Windows	Red Hat Linux
Teradata Database	17.x	✓	✓

ODBC



The ODBC and ODBC with CDC source endpoints can be used to access sources that are not included in Qlik Replicate's extensive Support Matrix. However, unless enabled through Professional Services and approved by your Account Manager, ODBC source endpoint types should not be used. Additionally, ODBC source endpoint types should not be used to access any sources already listed in the Support Matrix.

I Supported Platforms and Endpoints

Supported ODBC sources

Endpoint	Supported Versions	Windows	Red Hat Linux
ODBC	3.0 and 3.5	✓	✓
ODBC with CDC	3.0 and 3.5	✓	✓

NoSQL


Supported NoSQL sources

Endpoint	Supported Versions	Windows	Red Hat Linux
MongoDB (Standard)	4.x (starting from 4.2), 5.x, and 6.x	✓	✓

Other

Other supported sources

Endpoint	Supported Versions	Windows	Red Hat Linux
SAP Application	Supported backend endpoints: <ul style="list-style-type: none">• Microsoft SQL Server• Oracle• IBM DB2 for LUW• SAP HANA See Relational Databases above for version and platform information.	✓	✓
SAP Application (DB)	Supported backend endpoints: <ul style="list-style-type: none">• Oracle• Microsoft SQL Server• IBM DB2 for LUW• IBM DB2 for z/OS• SAP HANA See Relational Databases above for version and platform information.	✓	✓
SAP Extractor	N/A	✓	✓

Endpoint	Supported Versions	Windows	Red Hat Linux
SAP ODP	<p>The SAP ODP endpoint uses ODP API 2.0 which is available from the following SAP basis levels and above:</p> <ul style="list-style-type: none"> PI_BASIS 730 SP 14 (part of SAP NetWeaver 7.30 SP 14) PI_BASIS 731 SP 16 (part of SAP NetWeaver 7.03 SP 16 and 7.31 SP 16) PI_BASIS 740 SP 11 (part of SAP NetWeaver 7.40 SP 11) SAP_BW 750 SP 0 (incl. former PI_BASIS packages) <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;">  <i>SAP_BASIS and PI_BASIS are interchangeable. In newer systems, the component name is SAP_BASIS.</i> </div> <p>For more information, see SAP Note 1931427.</p> <p>For information on how to release extractors so that they are available for the ODP API, see SAP Note 2232584.</p>	✓	✓

I.3 Supported target endpoints

This section lists the target endpoint versions supported by Qlik Replicate, according to endpoint type.

- *Cloud-based (page 1320)*
- *Streaming (page 1324)*
- *File-based (page 1324)*
- *Hadoop (page 1325)*
- *Data warehouses (page 1325)*
- *Relational databases (page 1325)*
- *ODBC (page 1326)*

Cloud-based

Cloud-based supported target endpoints

Endpoint	Supported Versions	Windows	Red Hat Linux
Amazon Aurora MySQL Supported via the MySQL target endpoint.	Same as the MySQL target endpoint.	✓	✓

I Supported Platforms and Endpoints

Endpoint	Supported Versions	Windows	Red Hat Linux
Amazon Aurora PostgreSQL Supported via the PostgreSQL target endpoint.	Same as the PostgreSQL target endpoint.	✓	✓
Amazon MSK	N/A	✓	✓
Amazon RDS for MariaDB Supported via the MySQL target endpoint.	Same as the MySQL target endpoint.	✓	✓
Amazon RDS for MySQL Supported via the MySQL target endpoint.	Same as the MySQL target endpoint	✓	✓
Amazon RDS for SQL Server Supported via the Microsoft SQL Server target endpoint.	Same as the Microsoft SQL Server target endpoint (starting from 2014)	✓	✓
Amazon RDS for Oracle Supported via the Oracle target endpoint.	Same as the Oracle target endpoint (starting from 11.2)	✓	✓
Amazon RDS for PostgreSQL Supported via the PostgreSQL target endpoint.	Same as the PostgreSQL target endpoint.	✓	✓
Amazon Redshift	N/A	✓	✓
Snowflake on AWS (S3 storage)	N/A	✓	✓
Snowflake on AWS (Snowflake storage)	N/A	✓	✓
Snowflake on Azure (Azure Blob storage)	N/A	✓	X
Snowflake on Azure (Snowflake storage)	N/A	✓	✓
Snowflake on Google (Google Cloud Storage)	N/A	✓	✓
Amazon S3	N/A	✓	✓
Amazon EMR	5.x (starting from 5.2.x) and 6.x	✓	✓
Google Cloud SQL for MySQL	Same as the MySQL target endpoint.	✓	✓
Google Cloud SQL for PostgreSQL	Same as the PostgreSQL target endpoint	✓	✓

I Supported Platforms and Endpoints


Endpoint	Supported Versions	Windows	Red Hat Linux
Google Cloud AlloyDB for PostgreSQL	Same as the PostgreSQL target endpoint	✓	✓
Google Cloud Storage	N/A	✓	✓
Google Cloud BigQuery	N/A	✓	✓
Google Dataproc	1.x (starting from 1.2) and 2.x	✓	✓
Google Cloud SQL for SQL Server	Same as the Microsoft SQL Server target endpoint.	✓	✓
Microsoft Azure SQL Synapse Analytics (Blob storage)	N/A	✓	X
Microsoft Azure SQL Synapse Analytics (Data Lake Storage Gen2)	N/A	✓	✓
Microsoft Azure SQL Database	Same as the Microsoft SQL Server target endpoint.	✓	✓
Microsoft Azure SQL Managed Instance Supported via the Microsoft SQL Server target endpoint.	Same as the Microsoft SQL Server target endpoint.	✓	✓
Microsoft Azure Database for MySQL	Same as the MySQL target endpoint	✓	✓
Microsoft Azure Database for MySQL - Flexible Server Supported via the MySQL target endpoint.	N/A	✓	✓
Microsoft Azure Database for PostgreSQL	Same as the PostgreSQL target endpoint	✓	✓
Microsoft Azure Database for PostgreSQL - Flexible Server Supported via the PostgreSQL target endpoint.	N/A	✓	✓

I Supported Platforms and Endpoints

Endpoint	Supported Versions	Windows	Red Hat Linux
<p>Databricks (Cloud Storage)</p> <p>Databricks implementations supported via the Databricks (Cloud Storage) endpoint:</p> <ul style="list-style-type: none"> Databricks on AWS Databricks on Google Cloud Platform Microsoft Azure Databricks 	<ul style="list-style-type: none"> All-purposes (Interactive) cluster: <ul style="list-style-type: none"> 10.4 LTS 11.3 LTS 12.2 LTS SQL Warehouse cluster: N/A Databricks SQL Serverless 	✓	✓
Databricks Lakehouse (Delta)	<ul style="list-style-type: none"> All-purposes (Interactive) cluster: <ul style="list-style-type: none"> 10.4 LTS 11.3 LTS 12.2 LTS SQL Warehouse cluster: N/A Databricks SQL Serverless 	✓	✓
Netezza Performance Server (NPS)	N/A	✓	X
Microsoft Azure ADLS (Data Lake Storage Gen1 or Data Lake Storage Gen2)	N/A	✓	✓
Microsoft Azure HDInsight (Blob, Data Lake Storage Gen1, or Data Lake Storage Gen2)	4.x and 5.x	✓	✓
Oracle Autonomous Data Warehouse Supported via the Oracle target endpoint.	Same as the Oracle target endpoint (starting from Oracle 18.x)	✓	✓
Oracle on Oracle Cloud Supported via the Oracle target endpoint.	Same as the Oracle target endpoint (starting from Oracle 12.2)	✓	✓
Teradata Vantage Supported via the Teradata target endpoint.	17.x	✓	✓

Streaming

Streaming supported target endpoints

Endpoint	Supported Versions	Windows	Red Hat Linux
Kafka Certified with the following Schema Registry servers: <ul style="list-style-type: none"> • Confluent Cloud • Confluent Schema Registry 5.x (starting from 5.4), 6.x, and 7.x • Cloudera Schema Registry (based on Hortonworks) 	1.x, 2.x, 3.x	✓	✓
Confluent Cloud for Kafka Supported via the Kafka target endpoint <div style="border: 1px solid #ccc; padding: 5px; margin: 5px 0;">  <p><i>Although the Kafka target endpoint supports working with Confluent Cloud, customers who are not already using it this way should use the dedicated Confluent Cloud target endpoint instead.</i></p> </div>	N/A	✓	✓
Confluent Cloud	N/A	✓	✓
Microsoft Azure Event Hubs	N/A	✓	X
Amazon Kinesis Data Streams	N/A	✓	✓
Google Cloud Pub/Sub	N/A	✓	✓
MapR Streams	Please contact your Qlik Account Manager for support details.	X	✓

File-based

File-based supported target endpoints

Endpoint	Supported Versions	Windows	Red Hat Linux
File	N/A	✓	✓
File Channel	N/A	✓	✓

Hadoop

Hadoop supported target endpoints

Endpoint	Supported Versions	Windows	Red Hat Linux
Hadoop - Cloudera	7.x	✓	✓
Hadoop - Hortonworks	3.x (starting from 3.1)	✓	✓
Hortonworks Data Platform (HDP)	3.x (starting from 3.1)	✓	✓
Cloudera Data Platform (CDP) Private Cloud	7.x	✓	✓

Data warehouses

Data warehouses supported target endpoints

Endpoint	Supported Versions	Windows	Red Hat Linux
Action Vector	6.0	✓	✓
Vertica	11.x	✓	✓
Microsoft APS PDW	AU2	✓	X
Pivotal Greenplum	5.x, and 6.x	✓	✓
SAP Sybase IQ	15.x and 16.x	✓	X
Teradata Database	17.x	✓	✓
SingleStore	6.0.18	✓	X

Relational databases

Relational databases supported target endpoints

Endpoint	Supported Versions	Windows	Red Hat Linux
IBM DB2 for z/OS	DB2: 12, 12.1, and 13.1 IBM z/OS: 2.3, 2.4, and 2.5	✓	✓
Microsoft SQL Server	2014, 2016, 2017, 2019, and 2022	✓	✓
MySQL	5.7 and 8.0	✓	✓

Endpoint	Supported Versions	Windows	Red Hat Linux
MariaDB Supported via the MySQL target endpoint.	10.4, and 10.5	✓	✓
Oracle	11.x, 12.x, 18.x, 19.x, and 21.x	✓	✓
PostgreSQL	11.x, 12.x, 13.x, 14.x, and 15.x	✓	✓
SAP Sybase ASE	16.x	✓	X
SAP HANA	2.0	✓	✓

ODBC



The ODBC target endpoint, can be used to access targets that are not included in Qlik Replicate's extensive Support Matrix. However, unless enabled through Professional Services and approved by your Account Manager, the ODBC target endpoint should not be used. Additionally, the ODBC target endpoint should not be used to access any targets already listed in the Support Matrix.

ODBC supported target endpoints

Endpoint	Supported Versions	Windows	Red Hat Linux
ODBC	3.0 and 3.5	✓	✓

I.4 Endpoints supported in bidirectional replication

Bidirectional tasks support the following endpoints:

Source Endpoints:

- Amazon RDS for MySQL
- Amazon RDS for PostgreSQL
- Amazon RDS for SQL Server
- AWS Aurora Cloud for PostgreSQL
- File Channel
- Google Cloud SQL for MySQL

- Google Cloud SQL for PostgreSQL
- IBM DB2 for iSeries
- IBM DB2 for LUW
- IBM DB2 for z/OS
- Microsoft Azure SQL (MS-CDC)
- Microsoft SQL Server
- Microsoft SQL Server (MS-CDC)
- MySQL
- Oracle
- PostgreSQL
- SAP Sybase ASE

Target Endpoints:

- File Channel
- Google Cloud SQL for MySQL
- Google Cloud SQL for PostgreSQL
- IBM DB2 for z/OS
- Microsoft SQL Server
- Microsoft Azure SQL Database
- MySQL
- ODBC
- Oracle
- PostgreSQL
- SAP Sybase ASE

I.5 Compatibility with related products

Qlik Replicate May 2023 is compatible with the following related products only:

- Qlik Enterprise Manager May 2023 or later
- Compose May 2022 or later
- Compose for Data Lakes April 2020 (6.6)

I.6 Supported browsers

The following browsers are supported:

- Microsoft Edge (with automatic updates turned on)
- Mozilla Firefox (with automatic updates turned on)
- Google Chrome (with automatic updates turned on)



Displaying the console in a window that spans multiple vertical windows is not supported.

J Replicate Loggers

This appendix provides a description of the following Replicate loggers:

- *ADDONS* (page 1328)
- *ASSERTION* (page 1329)
- *COMMON* (page 1329)
- *COMMUNICATION* (page 1329)
- *DATA_RECORD* (page 1329)
- *DATA_STRUCTURE* (page 1329)
- *FILE_FACTORY* (page 1330)
- *FILE_TRANSFER* (AKA CIFTA) (page 1330)
- *INFRASTRUCTURE* (page 1330)
- *IO* (page 1330)
- *METADATA_CHANGES* (page 1330)
- *METADATA_MANAGER* (page 1330)
- *PERFORMANCE* (page 1330)
- *REST_SERVER* (page 1330)
- *SERVER* (page 1331)
- *SORTER* (page 1331)
- *SORTER_STORAGE* (page 1331)
- *SOURCE_CAPTURE* (page 1331)
- *SOURCE_LOG_DUMP* (page 1331)
- *SOURCE_UNLOAD* (page 1331)
- *STREAM* (page 1332)
- *STREAM_COMPONENT* (page 1332)
- *TABLES_MANAGER* (page 1332)
- *TARGET_APPLY* (page 1332)
- *TARGET_LOAD* (page 1332)
- *TASK_MANAGER* (page 1332)
- *TRANSFORMATION* (page 1333)
- *UTILITIES* (page 1333)

J.1 ADDONS

Only relevant when working with a Replicate add-on. Currently, the only add-ons are user-defined transformations.

J.2 ASSERTION

When the log contains an ASSERTION WARNING, it usually means that Replicate detected an anomaly with the data, which might result in replication issues at some stage. These warnings are not exposed in the web console and do not trigger notifications.

J.3 COMMON

Writes low level messages such as network activity.



Not recommended to set to "Trace" as it will write a huge amount of data to the log.

J.4 COMMUNICATION

Provides additional information about the communication between Replicate and the Source and Target components. For example, when using Hadoop, it will print the CURL debug messages and the "Apply" of the actual files (Mainly CURL and HTTP Client).

J.5 DATA_RECORD

Only available for some endpoints and may be implemented differently for each endpoint. It writes information about each change that occurs. While in Oracle it writes only the header fields, for some endpoints, such as Sybase ASE and IBM DB2 for LUW sources, it will also include the changed data. It records when a specific event was captured as well as the event context.

The content will not be presented; all events will be logged, even when a record is not in the task scope. (This is not relevant for Oracle LogMiner, as the list of objects are propagated to the LogMiner session).

Example

```
Produce INSERT event: object id 93685 context
'0000000003A4649301000001000005190000821F001000010000000003A46427' xid [000000000f9abd46]
timestamp '2017-06-07 09:01:58' thread 1 (oracdc_reader.c:2178)
```

J.6 DATA_STRUCTURE

Used for internal Replicate data structures and is related to how the code deals with the data and stores it in memory. In general, data structure is a way of organizing data in a computer so that it can be used efficiently.



Do not set to "Trace" unless specifically requested by Qlik.

J.7 FILE_FACTORY

Relevant to Hadoop Target, Amazon Redshift and Microsoft Azure SQL Synapse Analytics. This component is responsible for moving the files from Replicate to the target which, in the case of Hadoop, is the HDFS stage of the task.

J.8 FILE_TRANSFER (AKA CIFTA)

Writes to the log when the File Transfer component is used to push files to a specific location.

J.9 INFRASTRUCTURE

Records infrastructure information related to the infrastructure layers of Replicate code: ODBC infrastructure, logger infrastructure, PROTO_BUF, REPOSITORY, FILES USE, opening a new thread, closing a thread, saving the task state, and so on.

J.10 IO

Logs all IO operations (i.e. file operations), such as checking directory size, creating directories, deleting directories, and so on.

Example:

```
[IO ]T: scanning 'E:\Program Files\Attunity\Replicate\data\tasks\Task_name\data_files'  
directory size (at_dir.c:827)
```

J.11 METADATA_CHANGES

Will show the actual DDL changes which are included in the scope (available for specific endpoints).

J.12 METADATA_MANAGER

Writes information whenever Replicate reads metadata from the source or target, or stores it. Manage tables metadata, metadata store, and dynamic metadata.

J.13 PERFORMANCE

Currently used for latency only. Logs latency values for source and target endpoints every 30 seconds.

J.14 REST_SERVER

Handles all REST requests (API and UI). Also shows the interaction between Replicate and Qlik Enterprise Manager.

J.15 SERVER

The server thread in the task that communicates with the Replicate Server service on task start, stop, etc. Includes `init` functions for the task and the task definition.

J.16 SORTER

The main component in CDC that routes the changes captured from the source to the target.

Responsible for:

- Synchronizing Full Load and CDC changes
- Deciding which events to apply as cached changes
- Storing the transactions that arrive from the source database until they are committed, and sending them to the target database in the correct order (i.e. by commit time).

Whenever there is a CDC issue such as missing events, events not applied, or unacceptable CDC latency, it is recommended to enable "Verbose" for this logger.

J.17 SORTER_STORAGE

`SORTER_STORAGE` is the storage component of the Sorter which stores transactions (i.e. Changes) in memory and offloads them to disk when the transactions are too large, or unreasonably long. As this logger records a large amount of information, it should only be set to "Trace" if you encounter storage issues such as corrupt swap files, poor performance when offloading changes on disk, and so on.

J.18 SOURCE_CAPTURE

This is main CDC component on the source side. As such, it should be used to troubleshoot any CDC source issue. Note that setting to "Verbose" is not recommended unless absolutely necessary as it will record an enormous amount of data to the log.

Some target side components that use the source (e.g. LOB lookup) also use this logger. Setting the logger to "Trace" may be useful for troubleshooting performance issues and other source CDC issues such as bad data read from the source CDC.

J.19 SOURCE_LOG_DUMP

When using Replicate Log Reader, this component creates additional files with dumps of the read changes. The logger will write the actual record as it's being captured from the source.

Note that the data will be stored in a separate file and not in the log itself.

J.20 SOURCE_UNLOAD

Records source activity related to Full load operations and includes the `SELECT` statement executed against the source tables prior to Full Load.

J.21 STREAM

The Stream is the buffer in memory where data and control commands are kept. There are two types of stream: Data streams and Control streams. In the Data stream, source data is passed to the target or to the Sorter using this memory buffer. In the Control stream, Commands such as Start Full Load and Stop Full Load are passed to components.

As it records a large amount of data, this logger should only be set to "Trace" when a specific stream issue is encountered. Examples of stream issues include poor performance, issues with Control commands (e.g. commands not being performed), issues when loading many tables that may overload the control stream with Control commands, and so on.

J.22 STREAM_COMPONENT

Used by the Source, Sorter and Target to interact and communicate with the Stream component.

Example

```
Force switch to Transactional apply mode for Hadoop target (endpointshell.c:1340)
```

J.23 TABLES_MANAGER

Manage the table status including whether they were loaded into the target, the number of events, how the tables are partitioned, and so on.

J.24 TARGET_APPLY

Determines which changes are applied to the target during CDC and is relevant to both the Batch optimized apply and Transactional apply methods. It provides information about all Apply issues including missing events, bad data, and so on. As it usually does not record a lot of data, it can be safely set to "Verbose" in order to troubleshoot issues.

The logged messages will differ according to the target database.

J.25 TARGET_LOAD

Provides information about Full Load operations on the target side. Depending on the target, it may also print the metadata of the target table.

J.26 TASK_MANAGER

This is the parent task component that manages the other components in the task.

It is responsible for issuing commands to start loading or finish loading a table, create the different components threads, start or stop tasks, and so on.

It is useful for troubleshooting situations such as tables not loading, tables stuck in loading, one of the components not stopping or starting properly, and so on.

J.27 TRANSFORMATION

Logs information related to transformations. When set to "Trace", it will log the actual transformations being used by the task.

Example:

In the example below, a new column named "C" was added to the table. The expression is \$AR_H_STREAM_POSITION.

```
[TRANSFORMATION ]T: Transformation on table USER3.TEST1 exists (manipulation_manager.c:511)
[TRANSFORMATION ]T: Set transformation for table 'USER3.TEST1' (manipulator.c:596)
[TRANSFORMATION ]T: New column 'C', type: 'KAR_DATA_TYPE_STR' (manipulator.c:828)
[TRANSFORMATION ]T: Transformation expression is '$AR_H_STREAM_POSITION' (manipulator.c:551)
[TRANSFORMATION ]T: Final expression is '$AR_H_STREAM_POSITION' (expression_calc.c:822)
```

J.28 UTILITIES

In most cases, UTILITIES logs issues related to notifications.

K Replicate add-ons API

When developing add-on utilities that integrate with Replicate, best practice dictates that the utilities' memory requirements and logging capabilities should be managed by Replicate, rather than independently. To this end, the following SDK is provided. For information on Replicate add-on utilities, see *User-defined transformations* (page 1163) and *Using external credentials* (page 77).

All of the methods described in this API can be found in `<INSTALL_DIR>\addons\include\ar_addon.h`. To use these methods, you must reference this header file in your source code (`#include "ar_addon.h"`).

All methods will return one of the following values:

```
AR_ADDONS_STATUS_SUCCESS = 0,
```

```
AR_ADDONS_STATUS_FAILED = 1,
```

In this section

- *Memory management methods* (page 1334)
- *Log management methods* (page 1337)

K.1 Memory management methods

The following methods can be called to utilize Replicate's memory management procedures.


create_pool

Creates a new memory pool.

Syntax

```
create_pool (AR_AO_MPOOL *parentPool,
char *poolName,
AR_AO_MPOOL **newPool);
```

create_pool parameters

Parameter	Type	Description
parentPool	AR_AO_MPOOL	The pool to allocate from (required).
poolName	char	The new memory pool name (optional). <div style="border: 1px solid gray; padding: 5px; margin-top: 10px;">  <i>The poolName value is assumed to be a static, unmanaged space that does not need to be freed. The value will not be copied internally.</i> </div>
newPool	AR_AO_MPOOL	Returns the new memory pool.

destroy_pool

Deletes the given pool and deallocates the memory associated with it.



Using the `clear_pool` method (described below) will provide much better performance than using the `destroy_pool` and `create_pool` methods.

Syntax

```
destroy_pool (AR_AO_MPOOL *Pool);
```

destroy_pool parameters

Parameter	Type	Description
Pool	AR_AO_MPOOL	The pool to destroy.

clear_pool

Deallocates the memory associated with a given pool.

Syntax

```
clear_pool (AR_AO_MPOOL *pool);
```

clear_pool parameters

Parameter	Type	Description
pool	AR_AO_MPOOL	The pool to deallocate.

calloc

Performs the following operations:

- Allocates memory from a given pool



Different pools can be used to allocate memory on multiple threads concurrently. However, it's not thread safe to do this on the same pool.

- Allocates a buffer and sets all bytes to zero
- Aborts a process if no memory remains

Syntax

```
calloc (AR_AO_MPOOL *pool,
size_t *size
);
```

calloc parameters

Parameter	Type	Description
pool	AR_AO_MPOOL	The pool to allocate memory from.
size	size_t	The size of the buffer to allocate.

get_ctx_pool

Returns the ctx (context) thread pool (and creates a new pool if needed). Using ctx pool per thread minimizes the chances of a crash as each addon will have its own pool allocator. The pool will be destroyed automatically if a thread already exists.

Syntax

```
get_ctx_pool (char *addonName,
AR_AO_MPOOL *pool
);
```

get_ctx_pool parameters

Parameter	Type	Description
addonName	char	The pool name if one needs to be created (optional).
pool	AR_AO_MPOOL	The ctx pool that the function returns.

get_ctx

Gets the metadata from the thread pool.

Syntax

```
get_ctx_pool (AR_AO_MPOOL *pool,
char *key,
void **data
);
```

get_ctx parameters

Parameter	Type	Description
pool	AR_AO_MPOOL	The ctx pool to get the metadata from.
key	char	The key for the metadata to retrieve.
data	void	The user metadata associated with the pool(returned).

set_ctx

Sets metadata on the thread pool.



The data to be attached to the pool should have a life span at least as long as the pool to which it is being attached.

Syntax

```
set_ctx (AR_AO_MPOOL *pool,
char *key,
void *data,
AR_ADDONS_MEM_CTX_CLEANUP cleanup
);
```

set_ctx parameters

Parameter	Type	Description
pool	AR_AO_MPOOL	The ctx pool.
key	char	The key for the metadata to retrieve.
data	void	The user metadata to be associated with the pool.
cleanup	AR_ADDONS_MEM_CTX_CLEANUP	Cleans the data when pool is destroyed. Returns "0" on success.

K.2 Log management methods

The following methods can be used to write messages to the Replicate log files.



- To see messages in the log, you also need to set the "ADDONS" component (in the [task's Log Settings](#)) to the relevant logging level.
- Pointers should use %pp in the strFormat

log_error

Writes errors to the log.

Syntax

```
log_error (const char *strFormat, ...);
```

Parameters

log_error parameters

Parameter	Type	Description
strFormat	char	The error msg format.

log_warning

Writes warnings to the log.

Syntax

```
log_warning (const char *strFormat, ...);
```

Parameters

log_warning parameters

Parameter	Type	Description
strFormat	char	The warning message format.

log_trace

Writes trace messages to the log.

Syntax

```
log_trace(const char *strFormat, ...);
```

Parameters

log_trace parameters

Parameter	Type	Description
strFormat	char	The trace message format.

K Kafka Avro consumers API

Qlik Replicate supports writing messages to Kafka both in JSON format and in Apache Avro format. While JSON format is very easy to read, it is also rather verbose and slow to process. Apache Avro format, on the other hand, is extremely compact and can be processed very fast. However, using Avro for streaming presents a challenge in that Avro messages cannot be interpreted without the schema with which they were created. Moreover, Avro messages are usually encapsulated in a special "envelope" that points the processing agent to the correct schema for interpreting the message.

The advantages of Avro format over other plain formats such as JSON can be summed up as follows:

- More efficient in serialization space consumption
- Relatively easy to consume programmatically
- Supports long term schema evolution, which is important with data schemas that are likely to change over time

To ease the task of processing Avro messages, Qlik provides the Attunity Avro Message Decoder SDK, which contains the following:

- Avro message deserializer
- Message schema integration hooks
- Working samples

All Replicate message types covered in this section are encapsulated in a single message schema called the Replicate envelope. The schema of the Replicate envelope is as following:

```
{
  "type": "record",
  "name": "MessageEnvelope",
  "fields": [
    {"name": "magic", "type": {"type": "fixed", "name": "Magic", "size": 5}},
    {"name": "type", "type": "string"},
    {"name": "headers", "type": [{"null"}, {"type": "map", "values": "string"}]},
    {"name": "messageSchemaId", "type": [{"null"}, "string"]},
    {"name": "messageSchema", "type": [{"null"}, "string"]},
    {"name": "message", "type": "bytes"}
  ]
}
```

The fields in the envelope are as follows:

- `magic` (5 bytes fixed field)
The constant `"atMSG"` is used to identify this form of message. The `"atMSG"` constant should be used to validate that this message is indeed an Replicate envelope message.
- `type` (string field)
Describes the enveloped message type. This can be one of two values: `MD` which stands for metadata message and `DT` which stands for data message.
- `headers` (map of string key and value)
A free for use map for various properties set at the application level. Currently, no headers are set by Qlik Replicate but this may change in future versions.
- `messageSchemaId` (null or string)

A reference to a schema defined elsewhere, which can be used to deserialize the bytes in the message field. This specification does not explain how the schema ID is used for looking up the actual schema - it is an application level detail. This field is used exclusively with the `messageSchema` field.

- `messageSchema` (null or string)
An embedded UTF-8 encoded Avro JSON schema with which the message field can be serialized. This field is used exclusively with the `messageSchemaId` field.
- `message` (bytes)
An Avro encoded message, which is the payload of the message envelope.

Given the envelope schema, it is possible for anyone using this schema to properly decode the envelope messages from Kafka.

Once the envelope message has been decoded, there are two possible scenarios:

- **Scenario 1:** Decoding a self-describing message such as the metadata message
- **Scenario 2:** Decoding a message by referenced schema ID such as data messages

The method for logically decoding messages in both scenarios is described below.

K.3 Decoding a self-describing message

When the `messageSchema` field is not null, it means the `message` field can be decoded using the schema included in the `messageSchema` field. This is fairly straightforward to perform programatically since the only thing you need to usually supply Avro is a schema and a message, both of which are provided in the envelope message.

The Replicate metadata messages which include both table metadata, lineage and data schema description (to be referenced later by data messages) are enveloped in the self-describing envelope.

K.4 Decoding a message by referenced schema ID

Avro schemas are JSON documents which can be quite large, usually much larger than the data encoded by Avro conforming to the schema. For example, a schema of a 10 column table could be a JSON document of more than 100 characters while an actual row encoding of 10 columns may be only 10 bytes (depending of course on the type and length of fields). It is therefore typically not recommended to include schema and data together in a Kafka message because the schema information is redundant and is the same for all data messages while the actual data is the only thing which differs between data messages.

To avoid sending schema with each data message, each schema has a 32 bytes long ID. When a data message based on a previously sent data message schema (via the metadata message) is constructed, the `messageSchema` field is set to null and the `messageSchemaId` field is set to the 32 bytes ID of the schema instead. The application responsibility is to locate the data schema sent earlier in the metadata message and use that schema to decode the data message contained in the `message` field.

K.5 Typical consumer logic

A typical scenario involving Kafka involves Qlik Replicate as the Producer of messages into Kafka and customer code as the Consumer. Qlik Replicate offers the ability to define a specific topic as the schema topic and different topics for the table data.

The customer's consumer code should read metadata messages from the schema topic and then save the data schemas and any other information the consumer wishes to access later in a customer defined zone. Another set of customer consumers should read data messages from the various data topics, and access the data schemas zone as required to retrieve the data schemas required for decoding the data messages.

When consuming data messages and metadata messages from several topics and partitions in a multi-thread/process manner, a situation may arise where a given consumer may attempt to read a data message *before* the corresponding metadata message has been read. As it is not possible to read a data message before its corresponding metadata message, the consumer's logic should wait a reasonable amount of time until the corresponding metadata message has been read. If the metadata message is still not available after waiting for a reasonable amount of time, the consumer should handle this as an unexpected error and activate the planned error policy. An example of such a policy could be saving the message in a dedicated "delayed" topic for later processing.

As a rule of thumb, the number of metadata messages will be much lower (in the magnitude of 1:10000 or more) than the number of data messages. So, assuming a metadata consumer is active, the gap between metadata message and data message should be no more than a few seconds (usually, milliseconds).

See also:

- *Classes and Methods (page 1341)*
- *Sample Code Snippet (page 1346)*

K.6 Classes and Methods

To facilitate development (by the customer consumer developer(s)), Qlik provides a JAVA SDK packed as JAR for decoding its messages. The SDK is mainly built around the following classes/interfaces:

```
public interface MetadataStore {  
  
    byte[] loadMetadata(String schemaId);  
  
    void saveMetadata(String schemaId, byte[] data);  
}
```

This interface needs to be implemented by a customer class. Since the SDK cannot "guess" the behavior of the customer's code, the customer needs to supply a class implementing this interface.

The `loadMetadata` method locates data schemas (metadata) based on their ID.

The `saveMetadata` method stores the data schemas.

AttunityMessageDecoder

This class is used to decode messages read from Kafka:

```
public AttunityMessageDecoder(MetadadataStore metadadataStoreImplementor)
    throws AttunityDecoderException
```

A `MetadadataStore` must be passed to the constructor, otherwise an exception will be thrown.

Methods

- `public AttunityMessage decode(byte[] message)` throws `AttunityDecoderException` - This is the main method to call where the byte array input parameter is the Kafka message bytes read from the Kafka topic. The method signature returns a common parent class named `AttunityMessage` but in fact it will return only child object instances of this parent which are the `AttunityMetadadataMessage` class and the `AttunityDataMessage` class.
- `public static String getVersion()` - This method returns the Qlik Avro Decoder SDK version in the following format:
`major_version.minor_version.sp_version.build_id`
Example:
`1.0.0.3153`

AttunityMetadadataMessage

This class will be returned from the decoder SDK if the message is a metadata message.

Methods

The following methods return useful information:

- `public String getSchemaId()`
Returns the 32 bytes ID of the data schema described by this metadata message
- `public AttunityTableLineage getLineage()`
Returns table lineage information (class described later)
- `public String getDataSchema()`
Returns the data schema required to decode data schemas conforming to the schema ID of this message
- `public AttunityTableColumn[] getTableColumns()`
Returns information about the table columns included in this metadata message and subsequent data messages
- `public Map<String, String> getHeaders()`
Inherited from its parent class (`AttunityMessage`), this method will return the headers included in the envelope message
- `public GenericRecord getRawMessage()`
Inherited from its parent class (`AttunityMessage`), this method returns the raw decoded message as a Java Generic Record. This is often useful in the following cases:
 - When the metadata message needs to be serialized for future use.
 - When the existing SDK is older than the Qlik Replicate version writing the messages. In this case, some fields may not be returned by the methods described above and will only be accessible via the Generic Record interface.

AttunityDataMessage

This class will be returned from the decoder SDK if the message is a data message.

Methods

The following methods return useful information:

- `public String getSchemaId()`
Returns the 32 bytes ID upon which the data message is based.
- `public AttunityDataHeaders getDataHeaders()`
Returns data specific headers such as Transaction ID and Time. See also *AttunityDataType* (page 1344) below.
- `public AttunityTableLineage getLineage()`
Returns table lineage information (class described later)
- `public AttunityDataColumn[] getDataColumns()`
Returns the actual data columns as an array.
- `public AttunityDataColumn[] getBeforeDataColumns()`
If the operation described in the data message is an update operation, this method returns the row columns *before* the change while the `getDataColumns` method returns the row columns *after* the change.
As in the metadata class, the `getHeaders` and the `getRawMessage` methods are available as well as they are inherited from the same parent class (*AttunityMessage*). The usefulness of these methods is the same as in the metadata class.

AttunityTableLineage

This class is part of the *AttunityMetadataMessage* class.

Methods

- `public String getServer()`
Returns the name of the Replicate Server performing the data replication.
- `public String getTaskName()`
Returns the name of the Replicate task that is creating the messages.
- `public String getSchemaName()`
Returns the source database schema name (e.g. dbo).
- `public String getTableName()`
Returns the source table name.
- `public long getTableVersion()`
Returns the table version number. The number starts with "0" when the task starts and is incremented each time a DDL change occurs. Note that if the task is restarted in Full Load mode, the number will be reset to "0".
- `public Date getTimestamp()`
Returns the date/time when this message was created on the Replicate Server. This can be used to help track DDL changes in the source database.

AttunityTableColumn

This class is part of the *AttunityMetadataMessage* class.

Methods

- `public String getName()`
Returns the column name.
- `public AttunityAvroValueType getValueType()`
Returns a value type, describing the type of the column. This is not the original type on the source but rather the type forced by AVRO encoding supported types.
The possible values are:
BOOLEAN, INT32, INT64, FLOAT, DOUBLE, BYTES, and STRING
- `public AttunityDataType getDataType()`
Returns the `AttunityDataType` class (see description below).
- `public int getOrdinal()`
Returns the ordinal position of the column.
- `public int getPrimaryKeyPosition()`
Returns the position of table's primary key.

AttunityDataType

This class is part of the `AttunityTableColumn` class.

Methods

- `public AttunityDataValueType getValueType()`
Returns the Qlik Replicate data type.
Possible values are:
DATE, TIME, DATETIME, BYTES, BLOB, REAL4, REAL8, INT1, INT2, INT4, INT8, UINT1, UINT2, UINT4, UINT8, NUMERIC, STRING, WSTRING, CLOB, NCLOB, and BOOLEAN
- `public int getLength()`
Returns the length of the data column.
- `public int getPrecision()`
Returns the precision of the data column.
- `public int getScale()`
Returns the scale of the data column.
- `public boolean isNullable()`
Returns a boolean value if the column is nullable.

AttunityDataHeaders

This class is part of the `metadata` class.

Methods

- `public AttunityDataOperation getOperation()`
Returns the operation type performed on the row described by this data message. The possible operations are INSERT, UPDATE, DELETE, and REFRESH

Note that REFRESH describes the Full Load replication mode while the other operation types describe Change Processing operations.

- `public String getChangeSequence()`
Returns the operation's source change sequence.
- `public Date getTimestamp()`
Returns the date/time of the data operation.
- `public String getStreamPosition()`
Returns the operation's source stream position.
- `public String getTransactionId()`
Returns the operation's source transaction ID.
- `public byte[] getChangeMask()`
Returns which data columns were changed in the source table.
The change mask is a bitmask of data columns in little-endian order. The bit position in the change mask is based on the ordinal of the column in the metadata message of that table.
This means that if there are 10 data columns, they occupy bits 0 to 9 in the bitmask.
If UPDATE mask is 0B hexadecimal, which is 1011 binary – it means that the columns at ordinals 1, 2 and 4 were changed.
The following describes the bit semantics:
 - For INSERT records, all non-null columns have the associated bits set.
 - For DELETE records, only primary-key (or unique index) columns have the associated bits set. This allows an applier to construct a DELETE statement without having to find the primary key fields from another source.
 - For UPDATE records, each column with a changed value will have the associated bit set.
- `public byte[] getColumnMask()`
Returns which data columns are present in the message. Usually, this will include all of the table columns.



When replicating from an Oracle source without full supplemental logging, some columns might not be present in the data, since they could not be replicated.

The column mask is a bitmask of data columns in little-endian order. The bit position in the column mask is based on the ordinal of the column in the metadata message for that table.

This allows the applier to distinguish a null value that is the *actual* value of the column, from a null value that represents a column which could not be replicated from the source database.

AttunityDecoderException

This exception class can be thrown by the decode method in the `AttunityMessageDecoder` class.

Methods

The exception class has two public methods:

- `public String getErrorMessage()`
A formatted error message describing the error.
- `public AttunityDecoderExceptionCode getErrorCode()`

Returns the internal error code, which can be one of the following:

- `BAD_INPUT_MESSAGE`
The input message could not be parsed with the Qlik envelope schema.
- `BAD_MAGIC_VALUE`
The `magic` field in the Qlik envelope does not match the expected `magic` value (`atMSG`).
- `BAD_METADATA_MESSAGE`
The envelope was parsed successfully, but the metadata message could not be parsed using the embedded schema.
- `BAD_DATA_MESSAGE`
The envelope was parsed successfully, but the data message could not be parsed using the user supplied schema.
- `NO_METADATA_FOUND_IN_METADATA_STORE`
The envelope was parsed successfully and a data message was found, but calling the user supplied metadata store did not find a schema matching the data message ID.
- `UNRECOGNIZED_MESSAGE`
The envelope was parsed successfully, but the message type is not recognized by this module.
- `FAILED_TO_SAVE_INTO_METADATA_STORE`
The metadata message was created successfully, but saving bytes to the metadata store failed.
- `FAILED_TO_LOAD_FROM_METADATA_STORE`
The Envelope was parsed successfully, but loading bytes from the metadata store failed.

K.7 Sample Code Snippet

`AttunityMessageDecoder messageDecoder` – Previously defined and initialized with a proper schema locator.

`KafkaConsumer consumer` – Previously defined and initialized, and subscribed to some Kafka topic(s).

```

public void run()
{
    log("starting message consumption" , true , true);

    while (true)
    {
        ConsumerRecords<String, byte[]> records = consumer.poll(10000);
        if (records.isEmpty()) {
            continue;
        }

        log(String.format("got " + records.count() + " records") , true , true);

        for ( ConsumerRecord<String , byte[]> record : records) {
            try {
                AttunityMessage message = messageDecoder.decode(record.value());
                switch(message.getClass().toString())
                {
                    case "class com.attunity.avro.decoder.AttunityMetadataMessage" :
                        try {
                            handleMetadataMessage((AttunityMetadataMessage)message);
                        } catch (IOException e) {
                            e.printStackTrace();
                        }
                        break;

                    case "class com.attunity.avro.decoder.AttunityDataMessage" :
                        handleDataMessage((AttunityDataMessage)message);
                        break;

                    default:
                        log(String.format("Got unknown class type <%s> from Qlik
message decoder" , message.getClass().toString()) , true , true);
                        break;
                }
            } catch (AttunityDecoderException e1) {
                log(String.format("Error code is %s , reason is %s",
e1.getErrorCode(), e1.getErrorMessage()) , true , true);
            }
        }
    }
}

```

The code follows the following flow:

1. A Kafka consumer polls for messages. After the consumer has read the messages, each message is sent to the message decoder for decoding and message type verification.
2. If the message is of type `AttunityMetadataMessage`, a user-defined `handleMetadataMessage` method is called with the proper casting on the message.
3. If the message is of type `AttunityDataMessage`, a user-defined `handleDataMessage` method is called with the proper casting.
4. If an unknown message class is returned (i.e. the SDK supports more class types than the customer's code), an appropriate message is sent.
5. If decoding failed, an appropriate message is also sent, together with the exception details from the decoder.

L Glossary

c

Change Data Capture (CDC)

Captures changes in the source data or metadata as they occur and applies them to the target endpoint as soon as possible, in near-real-time. The changes are captured and applied as units of single committed transactions and several different target tables may be updated as the result of a single source Commit. This guarantees transactional integrity in the target endpoint. The CDC process for any file or table starts as soon as the data Load operation for the file or table begins.

F

Full Load

Creates all defined files or tables at the target endpoint, automatically defines the metadata that is required at the target, and populates the tables with data from the source.

L

Latency

Latency can be understood as follows: - Source Latency: The gap in seconds between the original change in the source endpoint and capturing it. - Target Latency: The gap in seconds between the original change in the source endpoint and applying it to the target endpoint. - Apply Latency: The gap in seconds between capturing the change in the source endpoint and applying it to the target endpoint.

s

Source Endpoint

A collection of files or tables managed by an endpoint management system (such as, Oracle, SQL Server) that is part of the main computing service of the IT organization of an enterprise. This source continuously updated, may need to provide a high throughput rate, may have strict 24/7 up-time requirements, and

may reference or update a number of tables in the course of a single logical transaction while providing transactional consistency and integrity for the data.

T

Target Endpoint

A collection of files or tables managed by an Endpoint Management System (DBMS), which may be different from the DBMS managing the source endpoint. It contains data that is derived from the source. It may contain only a subset of the tables, columns, or rows that appear in the source. Its tables may contain columns that do not appear in the source but are transformations or computations based on the source data.