Attunity Compose User Guide, Version 3.1

Copyright © 2018 Attunity Ltd. All rights reserved.

Primary Author: Charlton Book

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, endpoints, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Attunity license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software—Restricted Rights (June 1987). Attunity Ltd., 70 Blanchard Road, Burlington, MA 01803

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

Attunity is a registered trademark of Attunity Ltd and/or its affiliates. Other names may be trademarks of their respective owners.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Attunity is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Attunity is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Attunity is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.
Contents

1 | Introduction
   Why Attunity Compose? ................................................................. 10
   Key Features ............................................................................. 11
   Attunity Compose Architecture .............................................. 11

2 | Attunity Compose Installation and Setup
   Preparing your System for Compose ........................................ 14
   Setting Up HTTPS for the Compose Console ....................... 16
   Replacing the Self-Signed Certificate on Windows .......... 18
   Installing Compose ................................................................. 19
   Setting the Hostname and Changing the HTTPS Port ....... 20
   Determining the Required Number of Database Connections .. 21
   Changing the Master User Password ................................. 21
   Running Tasks on a Remote Compose Server .................. 23
   Accessing Attunity Compose ................................................. 23

3 | Getting Started with Attunity Compose
   The Attunity Compose Setup Process .................................... 25
   Introducing the Attunity Compose Interface ....................... 25
   Defining an Attunity Replicate Task ...................................... 28
   Attunity Compose Tutorial ...................................................... 29

4 | Adding and Managing Projects
   Adding Projects ........................................................................ 35
   Managing Projects .................................................................... 36
   Editing the Project Settings .................................................... 37
   Resetting Projects .................................................................... 40
   Project Deployment ................................................................. 40
      Exporting and Importing Projects using the CLI ................ 42
      Connecting to Attunity Compose Server ......................... 43
      Exporting a Project .............................................................. 43
      Importing a Project .............................................................. 45
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exporting the Project Configuration</td>
<td>46</td>
</tr>
<tr>
<td>Importing the Project Configuration</td>
<td>47</td>
</tr>
<tr>
<td>Moving Projects from the Test Environment to the Production Environment</td>
<td>48</td>
</tr>
<tr>
<td>Exporting Project Documentation</td>
<td>49</td>
</tr>
<tr>
<td>Viewing and Downloading DDL Scripts</td>
<td>50</td>
</tr>
<tr>
<td>Project Versioning</td>
<td>51</td>
</tr>
<tr>
<td>5</td>
<td>Setting up a Data Warehouse Connection</td>
</tr>
<tr>
<td>Adding a Data Warehouse in Compose</td>
<td>55</td>
</tr>
<tr>
<td>Using Oracle as a Data Warehouse</td>
<td>59</td>
</tr>
<tr>
<td>Prerequisites</td>
<td>55</td>
</tr>
<tr>
<td>Client</td>
<td>55</td>
</tr>
<tr>
<td>Permissions</td>
<td>56</td>
</tr>
<tr>
<td>Data Types</td>
<td>57</td>
</tr>
<tr>
<td>Defining the Connection Parameters</td>
<td>58</td>
</tr>
<tr>
<td>Using Microsoft SQL Server as a Data Warehouse</td>
<td>59</td>
</tr>
<tr>
<td>Prerequisites</td>
<td>59</td>
</tr>
<tr>
<td>Client</td>
<td>59</td>
</tr>
<tr>
<td>Permissions</td>
<td>59</td>
</tr>
<tr>
<td>Working with Windows Authentication</td>
<td>59</td>
</tr>
<tr>
<td>Data Types</td>
<td>60</td>
</tr>
<tr>
<td>Defining the Connection Parameters</td>
<td>60</td>
</tr>
<tr>
<td>Using Teradata as a Data Warehouse</td>
<td>62</td>
</tr>
<tr>
<td>Prerequisites</td>
<td>62</td>
</tr>
<tr>
<td>Client</td>
<td>62</td>
</tr>
<tr>
<td>Permissions</td>
<td>62</td>
</tr>
<tr>
<td>Data Types</td>
<td>63</td>
</tr>
<tr>
<td>Defining the Connection Parameters</td>
<td>63</td>
</tr>
<tr>
<td>Using Amazon Redshift as a Data Warehouse</td>
<td>65</td>
</tr>
<tr>
<td>Prerequisites</td>
<td>65</td>
</tr>
<tr>
<td>Permissions</td>
<td>66</td>
</tr>
<tr>
<td>Setting up SSL</td>
<td>66</td>
</tr>
<tr>
<td>Data Types</td>
<td>67</td>
</tr>
<tr>
<td>Defining the Connection Parameters</td>
<td>67</td>
</tr>
<tr>
<td>Managing Databases</td>
<td>69</td>
</tr>
</tbody>
</table>
6 | Setting up Data Source Connections

Data Source Permissions ................................................................. 70
  Source Landing Area Permissions ............................................... 70
  Source Database Permissions ..................................................... 71
Adding Data Sources in Compose .................................................... 71
Using Oracle as a Source .............................................................. 72
  Prerequisites ............................................................................. 72
  Data Types ............................................................................... 73
    Non-Supported Data Types ....................................................... 73
  Defining the Connection Parameters ........................................... 74
Using Microsoft SQL Server as a Source .......................................... 75
  Prerequisites ............................................................................. 75
  Working with Windows Authentication ......................................... 76
  Data Types ............................................................................... 76
    Non-Supported Data Types ....................................................... 77
  Defining the Connection Parameters ........................................... 77
Using MySQL as a Source ............................................................... 79
  Prerequisites ............................................................................. 80
    Cluster Prerequisites ............................................................. 80
  Data Types ............................................................................... 80
    Non-Supported Data Types ....................................................... 81
  Defining the Connection Parameters ........................................... 82
Using IBM DB2 as a Source ............................................................ 84
  Prerequisites ............................................................................. 84
  Data Types ............................................................................... 84
  Defining the Connection Parameters ........................................... 85
Using Teradata as a Source ............................................................. 87
  Prerequisites ............................................................................. 87
  Data Types ............................................................................... 87
  Defining the Connection Parameters ........................................... 88
Using Amazon Redshift as a Source ................................................ 90
  Prerequisites ............................................................................. 90
    Setting up SSL ....................................................................... 91
  Data Types ............................................................................... 92
  Defining the Connection Parameters ........................................... 92
Managing Databases ...................................................................... 94
# 7 | Creating and Managing the Model

Generating the Model ............................................................................................................. 95
  - Discovering the Source Database or Landing Area .......................................................... 95
  - Importing Entities and Mappings from Another Project .................................................. 98
  - Importing the Model from ERwin ..................................................................................... 99
    - Managing Global Mappings ......................................................................................... 99
Limitations .................................................................................................................................. 101
Validating the Model .................................................................................................................. 102
Displaying the Model ................................................................................................................ 102
Managing the Model .................................................................................................................. 105
  - The Manage Model Window ............................................................................................. 106
  - Managing Entities ............................................................................................................ 106
  - Managing Attributes ......................................................................................................... 107
    - Setting up Derived Attributes ....................................................................................... 110
    - Managing the Attributes Domain ................................................................................. 112
  - Creating and Managing Relationships .............................................................................. 113
    - Adding Relationships via the Manage Model Dialog Box ............................................. 113
    - Adding Relationships via the Display Model Dialog Box ............................................ 114
    - Deleting Relationships ................................................................................................... 115
  - Bulk Editing History Types and Satellite Numbers .......................................................... 115
  - Lineage and Impact Analysis ............................................................................................ 115
  - Adding Date and Time Entities to your Model .................................................................. 116
  - Designating Distribution Key and Primary Index Columns .......................................... 117
Creating Expressions ................................................................................................................ 118
Opening the Expression Builder ............................................................................................... 119
  - Expression Builder Overview .......................................................................................... 119
  - Building Expressions ........................................................................................................ 120
    - Parsing Expressions ...................................................................................................... 121
    - Editing Parameter Names ............................................................................................. 121
    - Testing Expressions ...................................................................................................... 122
Defining Reusable Transformations .......................................................................................... 124
  - Managing Reusable Transformations .............................................................................. 126

# 8 | Creating and Managing Command Tasks

Defining Command Tasks ......................................................................................................... 127
Managing Command Tasks ....................................................................................................... 128
## 9 | Creating and Managing the Data Warehouse

### Data Warehouse ETL Tasks ......................................................... 129
- How Compose Handles Missing References in the Data Warehouse ............................................. 129
- Creating the Data Warehouse Tables .................................................................................. 130
- Viewing the Data Warehouse Tables .................................................................................. 131
- Generating the ETL Set ....................................................................................................... 132
- Controlling Data Warehouse ETL Tasks .................................................................................. 133

### Managing ETL Sets .................................................................................. 134
- Adding and Duplicating ETL Sets ...................................................................................... 134
- Editing Column Mappings ..................................................................................................... 135
  - Adding, Deleting and Renaming Mappings .............................................................................. 139
  - Handling Duplicate Business Keys ...................................................................................... 139
- Handling Null Updates ........................................................................................................... 140
- Using Lookup Tables ............................................................................................................. 140
  - Using Lookup Tables that do not have an ETL Set for CDC Mapping ................................. 142
- Lookup Example .................................................................................................................... 142
- Creating Custom ETL Scripts .................................................................................................. 143
  - Single Table Example .......................................................................................................... 144
- Dropping and Recreating Tables ............................................................................................ 146
- Data Profiling ...................................................................................................................... 147
- Defining and Managing Data Quality Rules ........................................................................... 149
  - Defining Data Cleansing Rules ............................................................................................ 149
  - Defining Data Validation Rules .......................................................................................... 150
- Managing Data Quality Rules ................................................................................................ 151
- Viewing Missing References .................................................................................................... 152

### Viewing and Exporting ETL Commands .................................................. 153

### Modifying ETL Settings .................................................................................. 154

### Validating the Data Warehouse ............................................................................ 156

## 10 | Creating and Managing Data Marts

### Adding Data Marts and Star Schemas ......................................................... 158
- Understanding Star Schema Icons .................................................................................... 163
- Aggregation Example ............................................................................................................. 164

### Displaying Data in a Pivot Table ........................................................................ 165
- Pivot Table with Quarter Example ....................................................................................... 167
Managing Data Marts ........................................................................................................... 168
  Adding Star Schemas and Dimensions ............................................................................. 168
  Importing Dimensions ........................................................................................................ 169
  Editing Star Schemas .......................................................................................................... 170
  Setting Sort Keys ................................................................................................................ 172
  Editing Dimensions ............................................................................................................. 173
  Setting Sort Keys ................................................................................................................ 174
  Understanding Dimension History Types ............................................................................ 175
  Bulk Operations .................................................................................................................... 175
  Data Mart Views .................................................................................................................... 176
  Deleting Data Marts, Schemas and Dimensions ................................................................. 176

Pre and Post Loading ETLs .................................................................................................... 177
Viewing and Exporting ETL Commands .................................................................................. 178
Validating the Data Mart .......................................................................................................... 178
Modifying Data Mart Settings ................................................................................................ 180

11 | Controlling and Monitoring Tasks and Workflows

Viewing Information in the Monitor ..................................................................................... 184
Viewing Missing References .................................................................................................... 185
Controlling Tasks ..................................................................................................................... 187
  Running and Aborting Tasks Manually .............................................................................. 187
  Scheduling Tasks ................................................................................................................... 188
  Running ETL Tasks using the CLI ...................................................................................... 188
Defining Notifications Rules .................................................................................................. 189
  Managing Notification Rules ............................................................................................... 191
Workflows ............................................................................................................................... 191
  Creating Workflows ............................................................................................................. 192
    Continuing a Workflow in the Event of Parallel Task Failure ............................................ 193
  Validating Workflows .......................................................................................................... 194
  Managing Workflows .......................................................................................................... 194
  Running and Monitoring Workflows .................................................................................... 195
Monitoring and Controlling Replicate Tasks ............................................................................. 196

12 | Managing Compose

License Settings ....................................................................................................................... 198
  Registering a License .......................................................................................................... 198
    Register License ............................................................................................................... 199
# Table of Contents

- Viewing a License ................................................................. 200
- Logging Settings ................................................................. 200
  - Setting the Server Logging Level ........................................ 201
  - Setting Automatic Roll Over and Cleanup .......................... 201
  - Viewing and Downloading Compose Log Files .................... 202
- Mail Server Setting .............................................................. 203
- Running Tasks on a Remote Compose Server ......................... 203
- Monitoring and Controlling Replicate Tasks .......................... 204
- Setting up User Permissions .................................................. 205
  - Managing User Permissions .................................................. 206

**A | Impact of DST Change on Attunity Compose**

**B | Supported Platforms and Databases**

- Supported Windows Platforms ............................................. 209
- Supported Browsers ............................................................ 209
- Supported Data Sources ....................................................... 209
- Supported Data Warehouses ................................................. 210

**C | Cron Format and Examples**

- Cron Format ........................................................................... 211
- Special Characters .............................................................. 211
- Usage Examples .................................................................... 212

**D | Attunity Compose for Amazon Redshift Tutorial**

- PART 1: Satisfy the Prerequisites .......................................... 214
- PART 2: Launch the Attunity Compose for Amazon Redshift AMI ................................................ 216
- PART 3: Use Attunity Replicate to Move the Source Data to Amazon Redshift ................................................ 220
- PART 4: Use Compose to Create a Data Mart on Amazon Redshift ................................................ 223
- What Next? ........................................................................... 229

**Glossary**
1 | Introduction

The need for better decision-making is driving businesses to incorporate BI into day-to-day practices. BI is all about providing relevant information. For instance, if you know what your consumers are buying, you can develop products that match the current consumption trends and consequently boost profits. Likewise, BI enables you to identify key trends and patterns in your organization’s data and make the connections between important areas of your business that may otherwise seem unrelated.

However, setting up and maintaining a system that is capable of providing pertinent information on-demand is both costly and time-consuming. Enter Attunity Compose.

Attunity Compose automates the costly and time consuming process of designing, coding and continually updating your data warehouse and data marts.

In this chapter:
Why Attunity Compose?
Attunity Compose Architecture

Why Attunity Compose?

Agile Data warehouse automation (DWA) is a relatively new class of technology that accelerates warehouse development and change cycles while simultaneously assuring quality and consistency. Productivity gains, cost savings, and quality improvement are all possible with DWA.

Compose overcomes the complexity with automation, using significantly fewer resources at lower cost. It automatically designs, generates and populates enterprise data warehouses and data marts, adding data modeling and structuring capabilities inside the data warehouse. This unique technology is designed to provide fast, flexible delivery of information for Big Data Analytics and Business Intelligence (BI) projects such as:

» Building a new data warehouse
» Migrating an existing data warehouse to a new platform
» Expanding an existing data warehouse with new data feeds
» Generating data marts for lines of business
» Prototyping, performing ad hoc analytics and testing

By supporting the entire lifecycle of data warehouses and data marts through agile automation, Attunity Compose helps to eliminate traditional costly ETL development resources.
Key Features

» **Model-driven data warehouse automation** - The data warehouse is derived from a model that can be auto-generated from within Compose or imported from other tools such as ERwin. Thus, while Compose is model-driven, the model does not have to be prepared ahead of time. Rather, users can leverage Compose to reverse engineer sources for their model and then finalize their model as needed.

» **Automated management** of complex processes such as:
  » Schema generation for the Data Warehouse and Data Marts based on the derived model
  » Data integration (ETL)
  » Data quality and integrity
  » Logging
  » Maintenance and change management

» **Enterprise data warehouse support** - Full support for many enterprise data warehouses including Teradata, Oracle, Microsoft SQL Server and Amazon Redshift.

» **Data quality** - Compose lets you define data cleansing and validation rules that will be applied to the data before it is loaded into the data warehouse. In addition to defining rules, you can also define what action should be taken when data is rejected/accepted by a rule (Reject and Report, Reject Silently, and Reject and Abort).

» **Transformations** - Compose’s easy-to-use Expression Builder allows you to define transformations that filter and manipulate the extracted data according to the requirements of your data warehouse and data marts. Transformations can also be tested before being applied to actual data.

» **Monitoring** - ETL tasks can be monitored for errors using Compose’s integrated task monitor. The monitor provides detailed information about tasks including current status, start and end time, the number of rows inserted and updated, and the next scheduled task instance.

Attunity Compose Architecture

Attunity Compose’s automated model-driven technology saves time and money by quickly turning metadata into a streamlined data warehouse.
The process is illustrated in the following diagram and described below:

1. The source tables are loaded into the landing area using Attunity Replicate or a third-party replication tool. While the landing area is defined as the target in the replication task, it must be defined as the data source in a Compose task. The landing area can either be a schema or a database in the data warehouse (depending on how it is defined in the replication task).

   When using Attunity Replicate to move the source table to the landing area, you can define either a **Full Load** replication task or a **Full Load and Store Changes** replication task. Defining a **Full Load and Store Changes** replication task is recommended if you want to constantly propagate the source table changes to the data warehouse.

2. After the source tables are present in the landing area, Compose auto-generates a model based on the data source(s). Several methods are available for generating the model. For more information, see *Creating and Managing the Model*.

3. After the model has been created, you can extend the model with derived attributes, if needed.

4. Once the model has been finalized, instruct Compose to create the data warehouse tables (i.e. the staging tables and the actual data warehouse tables).

5. You can fine-tune the mappings between the tables in the landing area and the staging tables. You can also create custom ETLs and expressions as required. The Mappings ETL and the Custom ETLs are collectively referred to as the "ETL Set".

6. After the proper mappings are in place and the ETL Set has been generated, a single click is all it takes to extract the source data and load it into the data warehouse.

7. Once the data warehouse has been loaded, you can then select which tables to use as the fact, dimension and aggregation tables in your data mart(s).

8. When the data mart design is complete, Compose creates the data mart tables, generates the data mart ETLs and loads the data mart with the selected data.
It should be noted that even though setting up the initial project involves both manual and automatic operations, once the project is set up, you can automate the ETL tasks using the Compose scheduler.
This section describes how to install and set up Attunity Compose.

Note that as Attunity Replicate serves as a data (and metadata) provider for Attunity Compose, you also need to install Replicate in your organization. For a description of the Replicate installation procedure, refer to the Attunity Replicate User Guide and Reference.

**In this chapter:**
- Preparing your System for Compose
- Setting Up HTTPS for the Compose Console
- Installing Compose
- Setting the Hostname and Changing the HTTPS Port
- Determining the Required Number of Database Connections
- Changing the Master User Password
- Running Tasks on a Remote Compose Server
- Accessing Attunity Compose

### Preparing your System for Compose

Compose should be installed on a Windows Server machine that is able to access the data warehouse and optionally the source database(s) defined in your Compose project. Note that Compose only needs to access the source database if you plan to discover the source database when generating your model. For more information on discovery, see Discovering the Source Database or Landing Area.

For information on the supported databases and versions, see Supported Platforms and Databases.
Before installing Compose, make sure that the following prerequisites have been met:

- **Hardware configuration:**

  | Table 2.1 | Recommended Hardware Configuration |
  | --- | --- | --- |
  | Component | Basic System | Large System | Extra-Large System |
  | Processor | Quad core | Quad core base | 8-core base |
  | **Note** Additional cores may improve performance when several ETL processes are running concurrently. |
  | Memory | 8 GB | 16 GB | 32 GB |
  | **Note** Additional memory may improve performance when several ETL processes are running concurrently. |
  | Disk requirements | 100 GB SSD | 500 GB 10,000 RPM RAID | 500 GB 15,000 RPM RAID |
  | **Note** For all configurations, RAID is recommended for higher system availability in case of disk failure. |
  | Network | 1 Gb | 10 Gb | Two 10 Gb |

- **Ports:** The following firewall ports should be open on the Compose machine:
  - 80/443

- Microsoft Visual Studio C++ 2010 X64 Redistributable installed on the Compose machine.

- .NET Framework 4.5.2 or above installed on the Compose machine.

- **Supported Browsers:** The following browsers can be used to access the Compose Console (located on the Compose machine):
  - **Internet Explorer:** 11 and above
  - **Mozilla Firefox:** Latest version
  - **Google Chrome:** Latest version
  *Firefox and Chrome automatically update themselves to the latest version.*
Setting Up HTTPS for the Compose Console

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

As Compose 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 Compose 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 | find
tr /c:":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 Attunity Compose web user interface will automatically be available for secure access from a web browser using the following URL:

https://<machine-name>/AttunityCompose

Using the Self-Signed Certificate

Due to the way the HTTPS protocol works, there is no way for Compose to automatically provide and install a valid SSL server certificate. Still, in the event that no SSL server certificate is installed, Compose automatically generates and installs a self-signed SSL server certificate (as a temporary measure). This certificate is generated on the Compose 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 such as this one in Chrome:
Or this one in Firefox:

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.

**Note** 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:

```
composeCtl.exe certificate clean
```

**Note** that after the self-signed certificate is deleted, connections to the Attunity Compose machine will not be possible until a valid server certificate is installed. Should you want to generate a new self-signed certificate (to replace the deleted certificate), simply restart the Attunity Compose service.

### 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 Compose Server on Windows with their own certificate. The process, which is described below, involves removing the self-signed certificate and then importing the new certificate.

See also [Setting Up HTTPS for the Compose Console](index.html).

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 Attunity Compose:**

1. Stop the Attunity Compose service.
2. Open a command prompt (using the "Run as administrator" option) and change the path to the Compose bin directory. The default path is C:\Program Files\Attunity\Compose\bin.
3. Run the following command:
   ```
   composeCtl.exe certificate clean
   ```

**To import your own certificate:**

1. Run mmc.exe to open the Microsoft Management Console.
2. From the File menu, select Add/Remove Snap-in.
   The Add or Remove Snap-ins dialog box opens.
3. In the left pane, double-click Certificates.
   The Certificates snap-in wizard opens.
4. Select **Computer account** and then click **Next**.
5. In the **Select Computer** screen, make sure that **Local computer** is selected and then click **Finish**.
6. Click **OK** to close the **Add or Remove Snap-ins** dialog box.
7. In the left pane, expand the **Certificates** folder. Then, right-click the **Personal** folder and select **All Tasks > Import**.
8. In the **File to Import** screen, select your PFX certificate file. Note that by default the **Open** dialog box displays **CER** files. In order to see your PFX files, you need to select **Personal Information Exchange** from the drop-down list in the bottom right of the dialog box.
9. Click **Next** and enter the private key password.
10. Continue clicking **Next** until you reach the **Completing the Certificate Import Wizard** screen. Then click **Finish** to exit the wizard.
11. In the **Personal > Certificates** folder, double-click the newly imported certificate. The **Certificate** dialog box opens.
12. Scroll down the **Details** tab until you see the **Thumbprint** details and copy them to the clipboard.
13. Open a command prompt and run the following commands:
   **Syntax:**
   
   ◆ netsh http add sslcert ipport=0.0.0.0:443 certhash=[YOUR_CERTIFICATE-thumbprint WITHOUT_SPACES] appid={4dc3e181-e14b-4a21-b022-59fc669b0914}

   **Example:**
   
   netsh http add sslcert ipport=0.0.0.0:443
   certhash=5f6eccba751a75120cd0117389248ef3ca716e61 appid={4dc3e181-e14b-4a21-b022-59fc669b0914}

   **Syntax:**
   
   ◆ netsh http add sslcert ipport=[::]:443 certhash=[YOUR_CERTIFICATE-thumbprint WITHOUT_SPACES] appid={4dc3e181-e14b-4a21-b022-59fc669b0914}

   **Example:**
   
   netsh http add sslcert ipport=[::]:443
   certhash=5f6eccba751a75120cd0117389248ef3ca716e61 appid={4dc3e181-e14b-4a21-b022-59fc669b0914}

14. Close the command prompt and **Microsoft Management Console**.
15. Start the **Attunity Compose** service.

**Installing Compose**

The following section describes how to install **Attunity Compose**.
Note  For best performance when using Amazon Redshift as your data source or data warehouse, it is strongly recommended to install Attunity Compose on an Amazon EC2 machine located in the same region as your Amazon Redshift instance.

To install Compose:
1. Run the Compose setup file (Attunity_Compose_<version.build>.exe).
   The Attunity Compose setup wizard opens.
2. Click Next. Select I accept the terms of the license agreement and then click Next again.
3. Optionally change the installation directory and then click Next again.
4. Click Next and then click Next again to start the installation.
5. When the installation completes, click Finish to exit the Wizard.

Note  As part of the installation, a new Windows Service called Attunity Compose is created.

6. Open the Attunity Compose console as described in Accessing Attunity Compose.

Note  When you first open the Attunity Compose Console, you will be prompted to register an appropriate license. Register the license that you received from Attunity.

Setting the Hostname and Changing the HTTPS Port
After installing Attunity Compose, you can use the Compose CLI to set the hostname and HTTPS port for accessing the Attunity Compose server machine.

Under normal circumstances, you should not need to set the hostname. However, on some systems, connecting using HTTPS redirects to localhost. If this occurs, set the hostname of the Compose machine by running the command shown below.

To set the hostname:
Run the following command:

\[
<\text{product\_dir}/bin/\text{ComposeCtl.exe configuration set --address address}\n\]

where address is the hostname of the Compose server machine.

To change the HTTPS port:
Run the following command:

\[
<\text{product\_dir}/bin/\text{ComposeCtl.exe configuration set --https\_port port_number}\n\]
where `port_number` is the HTTPS port number of the Compose server machine. The default HTTPS port is 443.

Determining the Required Number of Database Connections

As a rule of thumb, the higher the number of database connections opened for Compose, the more tables Compose will be able to load in parallel. It is therefore recommended to open as many database connections as possible for Compose. However, if the number of database connections that can be opened for Compose is limited, you can calculate the minimum number of required connections as described below.

**To determine the number of required connections**

1. For each ETL set, determine the number of connections it can use during runtime. This value is specified in the Advanced tab in the data warehouse and data mart settings. When determining the number of required connections, various factors need to be taken into account including the number of tables, the size of the tables, and the volume of data. It is therefore recommended to determine the required number of connections in a Test environment.

2. Calculate the number of connections needed by all ETL tasks that run in parallel. For example, if three data mart ETL tasks run in parallel and each task requires 5 connections, then the number of required connections will be 15. Similarly, if a workflow contains two data warehouse ETL tasks that run in parallel and each task requires 5 connections, then the minimum number of required connections will be 10. However, if the same workflow also contains two data mart tasks (that run in parallel) and the sum of their connections is 20, then the minimum number of required connections will be 20.

3. Factor in the connections required by the Compose Console. To do this, multiply the maximum number of concurrent Compose users by three and then add to the sum of Step 2 above. So, if the number of required connections is 20 and the number of concurrent Compose users is 4, then the total would be:

   \[ 20 + 12 = 32 \]

Changing the Master User Password

All passwords are encrypted using a one-time randomly generated master key. The master key is stored automatically in the root repository of Compose (`<product_dir>/data/projects/GlobalRepo.sqlite`).

The master key is encrypted by a user key, which in turn, is derived from a master password entered by the user. By default, the Master User Password is randomly generated by Compose. The best practice, however, is to change the Master User
Password, as this will allow Compose projects and configuration settings to be imported to another machine without needing to re-enter the project credentials.

It may also be convenient to use the same Master User Password within a trusted environment. In other words, if the same administrators control both the production and the testing environments, using the same Master User Password in both environments will facilitate the transfer of projects with credentials between the testing and production environments.

The user key is stored in the muk.dat file located in <product_dir>\data\.

**Important:** The Master User Password must be a minimum of 32 characters. You can either use your own password or run the `genpassword` utility described below to generate a password for you. Note also that the password can only contain alphanumeric characters (i.e. it cannot contain special keyboard characters such as # or @).

All of the commands listed below must be run from: <product_dir>\Attunity\Compose\bin

**To generate a random 32 character password:**

Issue the following command:

```
ComposeCtl.exe utils genpassword
```

**To change the randomly generated master user password:**

1. Issue the following command:

```
ComposeCtl.exe masterukey set --password <new_master_password>
```

**Note** If you add the `--prompt` parameter to the command and omit the `--password` parameter, the CLI will prompt you for the password. When you enter the password, it will be obfuscated. This is especially useful if you do not want passwords to be retained in the command prompt history.

**Syntax:**

```
ComposeCtl.exe masterukey set --prompt
```

2. Restart the Compose service.

**To change a user-defined master user password:**

1. Issue the following command:

```
ComposeCtl.exe masterukey set --current-password <current_master_password> --password <new_master_password>
```
**Note** If you add the `--prompt` parameter to the command and omit the `--password` and `--current-password` parameters, the CLI will prompt you for the required passwords. When you enter the passwords, they will be obfuscated. This is especially useful if you do not want passwords to be retained in the command prompt history.

**Syntax:**
```
ComposeCtl.exe masterukey set --prompt
```

2. Restart the Compose service.

### Running Tasks on a Remote Compose Server

You can run Compose tasks either locally (the default) or on a remote Compose server.

**To run tasks on a remote server:**

1. From the Management menu in the projects view, select Java Server Settings.
2. In the Java Server Settings dialog box, select Remote Server and provide the required connection details.
3. Click OK to save your settings.

### Accessing Attunity Compose

You can use a Web browser to access the Attunity Compose Console from any computer in your network. For information on supported browsers, see Preparing your System for Compose.

**Note** The person logged in to the computer where you are accessing the Console must be an authorized Attunity Compose user. For more information, see Setting up User Permissions.

**To access the Attunity Compose Console:**

1. To access the Attunity Compose Console from the machine on which it is installed, select All Programs > Attunity Compose > Attunity Compose Console from the Windows Start menu. To access the Attunity Compose Console from a remote browser, type the following address in the address bar of your Web browser
   ```
   https://<computer name>/AttunityCompose
   ```
   where `<computer name>` is the name or IP address of the computer where Attunity Compose is installed.
2. If no server certificate is installed on the Compose machine, a page stating that the connection is untrusted will be displayed. This is because when Compose detects that no
server certificate is installed, it installs a self-signed certificate. Since the browser has no way of knowing whether the certificate is safe, it displays this page. For more information, see Setting Up HTTPS for the Compose Console.

3. When prompted for your password, enter your domain username and password.
3 | Getting Started with Attunity Compose

This section provides an overview of the Attunity Compose architecture, familiarizes you with its interface and ends with a short tutorial.

In this chapter:
- The Attunity Compose Setup Process
- Introducing the Attunity Compose Interface
- Defining an Attunity Replicate Task
- Attunity Compose Tutorial

The Attunity Compose Setup Process

Setting up Attunity Compose typically consists of the following stages (simplified):

1. In Attunity Replicate, define a task that replicates the source tables to a landing area in the data warehouse.
2. In Compose:
   a. Configure access to your data warehouse.
   b. Configure access to your data sources.
   c. Use the "Discover" option to auto-generate a model from the source tables or import an existing model that was created in ERwin. You can even create the model manually if you prefer.
   d. Create the data warehouse tables and populate them with the source data.
   e. Create a data mart from the data warehouse entities.
   f. Populate the data mart tables.

See also Attunity Compose Architecture.

Introducing the Attunity Compose Interface

This section will familiarize you with the elements that comprise the Attunity Compose UI.

To open Attunity Compose:

From the Windows Start menu, select All Programs > Attunity Compose > Attunity Compose Console.

The Attunity Compose Console opens in Management view.
Management View

In Management view, you can perform the following tasks:

- Create, edit, delete and duplicate projects
  For more information, see Adding and Managing Projects.
- Access Attunity Compose management options, including:
  - Register and view the product license
  - Manage log levels and cleanup options
  - Manage email settings
  For more information, see Managing Compose.

Designer View

When you add a new project or open an existing project, the console switches to Designer view. If you are in Monitor view (see below), you can switch back to Designer view by clicking the Designer tab in the top right of the console.
Figure 3.2 | Designer View

Designer view comprises the following panels:

- **Databases** - Configure access to your source database(s) and data warehouse.
  
  For more information, see Setting up Data Source Connections and Setting up a Data Warehouse Connection respectively.

- **Model** - Create and edit your model.
  
  For more information, see Creating and Managing the Model.

- **Data Warehouse** - Create the data warehouse tables, generate the ETL code, and run data warehouse ETL tasks.
  
  For more information, see Creating and Managing the Data Warehouse.

- **Data Mart** - Define data marts, create the data mart tables, generate the ETL code, and run data mart ETL tasks.
  
  For more information, see Creating and Managing Data Marts.

In Designer view, each of the panels has a bar below the panel name. The bar can be empty, half-filled or completely filled, according to the current configuration status of the panel properties, as follows:

- No fill (gray) - Not configured

- Half filled - Configuration is not complete
Monitor View

To switch to Monitor view, click the Monitor tab in the top right of the console.

**Figure 3.3 | Monitor View**

In Monitor view, you can view the status of data warehouse and data mart ETL tasks and schedule their execution, either individually or as a workflow.

For more information, see [Controlling and Monitoring Tasks and Workflows](#).

Defining an Attunity Replicate Task

In order to work with Compose, you first need to define an Attunity Replicate task that replicates the source tables from the source endpoint to a "landing area" in the data warehouse. The landing area should then be defined as the data source in the Compose project.

**Note** The steps below highlight the settings that are required when using Attunity Replicate with Compose. For a full description of setting up tasks in Attunity Replicate, please refer to the [Attunity Replicate User Guide and Reference](#).
To define the task:

1. Open Attunity Replicate and define a new task.
   - To enable Full Load and Change Processing replication, enable the Full Load and Store Changes options (the Apply Changes option should not be enabled).
   - To enable Full Load replication only, make sure that only the Full Load option is enabled.
   - To enable Change Processing replication only, make sure that only the Store Changes option is enabled. Note that this option should only be selected if the Full Load tables and data already exist in the landing area.
   - To enable change processing for lookup tables that already exist in the landing area and are not part of the Compose model, enable the Apply Changes option only. Note that such a task should be defined in addition to the Full Load and Store Changes replication task described above. For more information on updating standalone lookup tables, see Using Lookup Tables that do not have an ETL Set for CDC Mapping.

2. Open the Manage Endpoint Connections dialog box and define a source and target endpoint. The target endpoint must be the database where you want Compose to create the data warehouse.

3. Add the endpoints to the Replicate task and then select which source tables to replicate.

4. In the Task Settings’ Target Metadata tab, specify a Target table schema name.

5. Run the task.
   - Wait for the Full Load replication to complete and then continue the workflow in Compose as described in the Attunity Compose Tutorial below and in Adding and Managing Projects.

Attunity Compose Tutorial

This short tutorial will walk you through each of the stages required to create an Attunity Compose project. For simplicity’s sake, we will be using Microsoft SQL Server as both the source database server and the target Data Warehouse. You can of course use any of the supported source or target databases, but instructions for doing so are outside the scope of this tutorial.

What you need:

- Attunity Compose installed according to the instructions in Attunity Compose Installation and Setup.
- The Northwind.MDF sample database attached to Microsoft SQL Server.
  - An easy-to-follow set of instructions for downloading and installing Northwind.MDF can be found at the following website: http://businessimpactinc.com/install-northwind-database/
- Define an empty database on Microsoft SQL Server (e.g. northwind_dwh) and make a note of its name. This will serve as the target Data Warehouse for the Northwind.MDF
source tables.

» Microsoft SQL Server Native Client 11.0 installed on the Attunity Compose machine.

To set up a Compose project:

1. Define and run a replication task in Attunity Replicate as described in Defining an Attunity Replicate Task.
2. Open Attunity Compose.
3. Click the New Project toolbar button.
4. Enter a name for your project and then click OK.
5. In the Databases panel, perform the following steps to define your data warehouse:
   a. Click Manage.
      The Manage Databases dialog box opens.
   b. Click the Add New Database link or the New toolbar button.
      The New Data Warehouse window opens.
   c. In the New Data Warehouse window:
      » In the Name field, specify a display name for your data warehouse.
      » From the Type drop-down list, select Microsoft SQL Server.
      » In the Server Name field, specify the Microsoft SQL Server name using the following format:
         To connect to a named Microsoft SQL Server instance:
         computer_name\db_server_name
         To connect to the default Microsoft SQL Server instance:
         computer_name
      » In the User Name and Password fields, enter your credentials for logging in to the server specified in the Server Name field.
      » In the Database Name field, specify the name of the database specified in the target endpoint of the Attunity Replicate task.
      » In the Data Warehouse Schema field, specify dbo or your preferred schema.
      » In the Data Mart Schema field, specify dbo or your preferred schema.

     Note  You can specify different schemas for the data warehouse and data mart tables, but for the purpose of this quick start, we’ll use the same schema.

    » Click Test Connection to verify that Compose is able to establish a connection to the specified database and then click OK to save your settings.
   d. Click New.
      The New Data Source dialog box opens.
   e. In the New Data Source dialog box:
» In the Name field, specify a display name for your data source.
» From the Content Type drop-down list, choose Full Load and Change Processing.
» From the Designate By drop-down list, choose Schema.
» In the Schema name field, enter the schema name that you specified in the Target Metadata tab of the Replicate task. For more information, see Defining an Attunity Replicate Task.
» In the Error mart schema name field, specify the schema where you want the data mart exception tables to be created. Data that is rejected by data quality rules will be copied to tables in the specified schema.
» Select the Source database connection check box and then provide the details for connecting to the source database. For the purpose of this tutorial, except for the Schema, these should be the same as the data warehouse connection details.
» Click Test Connection to verify that Compose is able to establish a connection to the specified database and then click OK to save your settings.
» Click OK to save your settings.

6. In the Model panel, perform the following steps to create the model for data warehouse generation:
   a. From the drop-down menu in the top right corner of the Model panel, select Discover.
      The Discover dialog box opens.
   b. Select the source database (i.e. the database without the "_landing" suffix). This is the source endpoint in the Attunity Replicate task.
      The Source Table/View Selection - <Data_Source_Name>_Landing window opens.
   c. In the Source Table/View Selection window:
      » Select the Tables option.
      » Click the Search button.
      » From the Results list, select which tables to discover and then click OK.
      The Generating Model from <db_name> window opens. Wait for the model generation to complete and then click Close.

7. In the Data Warehouse panel, perform the following steps to populate the Data Warehouse with the source data:
   a. Click Create.
      The Creating Data Warehouse window opens. Wait for the Data Warehouse to be created and then click Close.
   b. Click Manage.
      The Manage ETL Set window opens.
   c. Click Generate.
The Generating Instructions for ETL Set: <Name> window opens. Wait for the ETL instruction set to be generated and then click Close.

d. Click Run.

The Manage ETL Sets window switches to Monitor view and Attunity Compose starts to populate the Data Warehouse with data (this may take a few seconds).

e. Wait for the Data Warehouse to be populated and then close the Manage ETL Sets window.

8. In the Data Mart panel, perform the following steps to create a data mart with a star schema:

a. Click New.

The New Data Mart dialog box opens. Leave the default name.

b. Make sure the Start New Star Schema Wizard check box is selected. Then click OK.

The New Star Schema wizard opens. Leave the default name.

c. Select Transactional as the star schema type and then click Next.

d. In the Facts screen, select Order Details. Then click Next.

e. In the Dimensions screen, clear all the check boxes and then select Customers and Products only, as shown below.

f. Then click Next.

g. In the Transaction Date screen, select OrderDate. Then click Finish.
The star schema is displayed on the right of the Manage Data Marts window.

h. Click Create Tables.
   The Creating Data Mart: <Data Mart Name> window opens. Wait for the Data Mart tables to be created and then close the window.

i. Click Generate.
   The Generating Instructions for Data Mart: <Data Mart Name> window opens. Wait for the generation of ETL Instructions to complete and then close the window.

j. Click Run.
   The Manage Data Marts window switches to Monitor view and Attunity Compose populates the Data Mart with data. Leave the Manage Data Marts window open in Monitor view for now (The two buttons at the top right of the window allow you to switch between Designer and Monitor views).

9. To display the data in a pivot table:
   a. Click the Pivot toolbar button.
      The Select Columns for Pivot Table window opens.
   b. From the drop-down list at the top of the window, select the Pivot Table columns as follows:
      » In the 1Fct_Order Details table, select Quantity.
      » In the 1Dim_Customers table, select Country.
      » In the 1Dim_Products table, select ProductName.
   c. Click OK.
      A Pivot Table is created with your selected columns.
   d. Drag the Quantity box to the space above the table and the ProductName box to the space on the left.
      Select Heatmap from the drop-down list below the Customize Columns button.
Your pivot table should now look like this:
This section describes how to add and manage Compose projects.

In this chapter:
- Adding Projects
- Managing Projects
- Editing the Project Settings
- Resetting Projects
- Project Deployment
- Exporting Project Documentation
- Viewing and Downloading DDL Scripts
- Project Versioning

Adding Projects
Adding a new project is the first task you need to undertake in order to work with Attunity Compose. You can set up as many projects as you need.

To create a new project:
1. Click the New Project toolbar button.
   The New Project dialog box opens.
2. Specify a name for your project and then click OK.

   Note  The following names are reserved system names and cannot be used as project names: CON, PRN, AUX, CLOCK$, NUL, COM1, COM2, COM3, COM4, COM5, COM6, COM7, COM8, COM9, LPT1, LPT2, LPT3, LPT4, LPT5, LPT6, LPT7, LPT8 and LPT9.

3. The project panels will be displayed.
4. Add at least one source database and a data warehouse as described in Setting up Data Source Connections and Setting up a Data Warehouse Connection respectively.
5. Create a model as described in Creating and Managing the Model.
6. Set up the data warehouse as described in Creating and Managing the Data Warehouse.
7. Set up the data mart as described in Creating and Managing Data Marts.

Managing Projects
You can edit, duplicate and delete projects as required. The table below describes the available options.

<table>
<thead>
<tr>
<th>To</th>
<th>Do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit a project</td>
<td>Any of the following:</td>
</tr>
<tr>
<td></td>
<td>» Double-click the project.</td>
</tr>
<tr>
<td></td>
<td>» Right-click the project and select Open.</td>
</tr>
<tr>
<td></td>
<td>» Select the project and then click the Open toolbar button.</td>
</tr>
<tr>
<td>Duplicate a project</td>
<td>Any of the following:</td>
</tr>
<tr>
<td>Duplicating a project</td>
<td>» Right-click the project and select Duplicate.</td>
</tr>
</tbody>
</table>

**Note** Project management operation can only be performed in the main Compose window. To switch to the main window, click the downward arrow to the right of the project name and then select All Projects from the drop-down menu.
saves time when you need to create multiple projects with similar settings.» Select the project and then click the Duplicate toolbar button.

**Note** After duplicating a project, you must edit the data warehouse connection settings to use a different database. Otherwise, every change in the original project will be reflected in the duplicated project and vice versa.

<table>
<thead>
<tr>
<th>Delete a project</th>
<th>Any of the following:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>» Right-click the project and select Delete.</td>
</tr>
<tr>
<td></td>
<td>» Select the project and then click the Delete toolbar button.</td>
</tr>
</tbody>
</table>

**Editing the Project Settings**

You can change the project settings according to your needs.

**To access the project settings:**

1. Open your project as described in Managing Projects.
2. Click the downward arrow to the right of the project name and select **Project Settings** from the drop-down menu.
   
   The **Settings** dialog box opens.
3. In the **General** tab, optionally select the following options:
   
   » **Generate DDL scripts but do not run them** - By default, Compose executes the CREATE, ADJUST and DROP statements immediately upon user request. When you select this option, Compose will only generate the scripts but not execute them. This allows you to review and edit the scripts before they are executed.
      
      For example, if you want your data warehouse tables to contain partitions, you will need to edit the CREATE statement to create the partitions. Another example is if you are using Teradata as your data warehouse and you want to change the Distribution Key set by Compose.
      
      Note that if you select this option, you will need to copy the scripts to your data warehouse and run them manually. You can view, copy and download the DDL scripts as described in Viewing and Downloading DDL Scripts.

   **Note** The UI is not refreshed after manually running the DDL scripts (when the "Generate DDL scripts but do not run them" option is selected). For example, if you run a script that creates the data warehouse tables, the UI will not show the tables until the display is refreshed.

   **Workaround:**
Press F5 to refresh the browser display.

- **Ignore Mapping Data Type Validation** - By default, Compose issues a validation error when a landing table is mapped to a staging table with a different data type. You can select this option to allow the mapping of different data types. Note that you should only select this option if you need to map landing table data types to compatible (though not identical) staging table data types.

- **Write metadata to the TDWM tables in the data warehouse** - When this option is selected (the default unless Amazon Redshift is the data warehouse), Compose writes the metadata for the data warehouse tables to the following tables:
  
  `<schema>.TDWM_Tables` and `<schema>.TDWM_COLUMNS`

  Centralizing the metadata in two dedicated tables makes it easier for external metadata tools to analyze the metadata. The metadata is also written to the local Compose repository, so clearing this option (if performance issues are encountered) will not affect Compose functionality in any way.

- **Do not display the default workflows in the monitor** - Select this option if you want to prevent the default workflows from being executed.

4. Optionally, change the settings in the **Names** and **Defaults** tabs according to the descriptions in the tables below.

**Note** If you change the prefix or suffix of existing tables (e.g. data warehouse tables), you need to drop and create the data warehouse and data mart tables.

### Table 4.1 | Names Tab

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>From Date Column Name</td>
<td>The name of the &quot;From Date&quot; column. This column is added to tables that contain attributes (columns) with a History Type 2. The column is used to delimit the range of dates for a given record version. This name cannot be used in other columns.</td>
</tr>
<tr>
<td>To Date Column Name</td>
<td>The name of the &quot;To Date&quot; column. This column is added to tables that contain attributes (columns) with a History Type 2. The column is used to delimit the range of dates for a given record version. This name cannot be used in other columns.</td>
</tr>
<tr>
<td>Replicate Change Table Suffix</td>
<td>The suffix used to identify Replicate Change Tables in the landing area of the data warehouse.</td>
</tr>
<tr>
<td>Archived Change Suffix</td>
<td>The suffix used to identify archived Change Tables in the specified database. For more information on archiving Change Tables, see <strong>After applying</strong></td>
</tr>
</tbody>
</table>
### Table 4.1 | Names Tab

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table Suffix</td>
<td>changes. Note that although the cross-referenced description is part of the Microsoft SQL Server source topic, this option is available for all data sources.</td>
</tr>
<tr>
<td>Data Warehouse Prefix</td>
<td>The prefix added to table names in the Data Warehouse.</td>
</tr>
<tr>
<td>Data Warehouse View Prefix</td>
<td>The prefix added to view names in the Data Warehouse.</td>
</tr>
<tr>
<td>HUB Suffix</td>
<td>The suffix used to identify hub tables in the Data Warehouse. Hub tables contain History Type 1 columns. History Type 1 column do not contain any version history as opposed to History Type 2 columns that do (see SAT below).</td>
</tr>
<tr>
<td>SAT (Satellite) Suffix</td>
<td>The suffix used to identify satellite tables in the Data Warehouse. Satellite tables contain History Type 2 columns. History Type 2 columns keep a history of the data version by adding a new row whenever the data is updated.</td>
</tr>
<tr>
<td>Data Mart Prefix</td>
<td>The prefix used to identify to tables in the Data Mart.</td>
</tr>
<tr>
<td>Exception Mart Suffix</td>
<td>The suffix used to identify error tables in the data warehouse. These tables contain data that was rejected by a data quality rule.</td>
</tr>
</tbody>
</table>

### Table 4.2 | Defaults Tab

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lowest Date</td>
<td>The value stored in the &quot;From Date&quot; column. This is the date when the version started.</td>
</tr>
<tr>
<td>Highest Date</td>
<td>The value stored in the &quot;To Date&quot; column. This is the date when the version ended.</td>
</tr>
<tr>
<td>Unknown Date</td>
<td>The value shown when a date is missing.</td>
</tr>
<tr>
<td>Unknown String</td>
<td>The value shown when a string cannot be interpreted.</td>
</tr>
<tr>
<td>Unknown Number</td>
<td>The value shown when a number cannot be interpreted.</td>
</tr>
<tr>
<td>Use Datetime2</td>
<td>Whether to use datetime2 or not. Although, offering more options, datetime2 is not supported by all Data Warehouses.</td>
</tr>
</tbody>
</table>
Resetting Projects

You can reset projects as required. This can be useful in the project development stage as it allows you to easily delete unwanted project elements. Be careful not to reset a project and delete data in a production environment!

To reset a project:

1. Open your project as described in Managing Projects.
2. Click the downward arrow to the right of the project name and select Reset Project from the drop-down menu.
   The Reset Project dialog box opens.
3. Choose to reset any of the following:
   » Model (Entities, Relationships, Attribute Domains).
      For more information on models, see Creating and Managing the Model.
   » Global Mappings
      For more information on global mapping, see Managing Global Mappings.
   » Data Warehouse and Data Marts
      For more information on data warehouses and data marts, see Creating and Managing the Data Warehouse and Creating and Managing the Data Warehouse respectively.
   » Delete DDL Scripts
      For more information on DDL scripts, see Editing the Project Settings and Viewing and Downloading DDL Scripts.
4. Click Reset Project and then click Yes when prompted to confirm your request.

Project Deployment

Project deployment packages can be used to back up projects or migrate projects between different environments (e.g. testing to production). As a deployment package is intended to be deployed in a new environment, it contains the data warehouse and data source definitions, but without any passwords. The deployment package also does not contain any data from the data warehouse or data mart, only the metadata. The deployment package also contains the project model and mapping information, which should be consistent with the landing area tables in the new environment.

For a complete list of objects contained in the deployment package, see Exporting Projects.

Creating Deployment Packages

This section explains how to create a project deployment package.
To create a deployment package

1. Choose one of the following methods:
   » In the main Compose window, right-click the desired project and select **Create Deployment Package** from the context menu.
   » In the main Compose window, select the desired project. Then, click the **Deployment** toolbar button and select **Create Deployment Package** from the drop-down menu.
   » In the project window, select **Deployment > Create Deployment Package** from the project drop-down menu.
   The **Create Deployment Package - <Project_Name>** window opens.

2. Provide a **Version** number and a **Description** in the designated fields and then click **OK**.

   A ZIP file containing a JSON file (i.e. the project settings) and a **readme.txt** file will be saved to your browser's default download location.

   The ZIP file name is in the following format:
   `<Project_Name>_deployment_<Date>_<Time>.zip`

   The **readme.txt** file contains the following information about the deployment package: project name, export date, exporter user name, deployment version, and description.
Deploying Packages

This section explains how to deploy a project deployment package. You can only deploy packages to an existing project. Therefore, before deploying a project, create a new project with the user name and password required for connecting to the data warehouse and source database (if defined) in the new environment. Note that any existing data mart schemas will not be overridden during project deployment.

Note: If preferred, you can create an empty project and provide the required credentials after the deployment completes. In this case, an error message prompting you for the missing credentials will be displayed after the deployment completes.

To deploy a project deployment package

1. Copy the ZIP file created in Creating Deployment Packages to a location that is accessible from the Compose machine.
2. Open Compose and choose one of the following methods:
   » In the main Compose window, select the desired project. Then, click the Deploy-ment toolbar button and select Deploy from the drop-down menu.
   » In the project window, select Deployment > Deploy from the project drop-down menu.
   The Deploy window opens.
3. Either drag and drop the file on the window.
   - OR-
   Click Select and browse to the location of the deployment package. In the Open dialog box, either double-click the deployment package ZIP file or select the file and click OK.
   The package details will be displayed.
4. Click Deploy to deploy the package. When prompted to replace the existing project, confirm the operation.
   The project will be deployed.

Exporting and Importing Projects using the CLI

Important: Compose CLI requires Administrator permission. To grant Administrator permission, select "Run as administrator" when opening the command prompt.

Under normal circumstances, use the deployment options described in Project Deployment to export and import projects. For deployment automation or control by another tool, you can use the command line interface (CLI) to perform the following tasks:

» Exporting a Project
» Importing a Project
Exporting the Project Configuration
Importing the Project Configuration

Note  To export or import a project or project configuration, you first need to change the default Master User Password.
For more information on changing the master user password, see Changing the Master User Password.
See also: Moving Projects from the Test Environment to the Production Environment and Import/Export Scenarios - When is a Password Required?

Before running any command, you must run the Connecting to Attunity Compose Server command.
To get help when using the command line, you can run the Help command. For example, for help about exporting a project, issue the following command:
ComposeCli.exe export_project_repository --help
This brings up a list of help parameters.

Connecting to Attunity Compose Server
Run the Connect command to establish a connection to the Attunity Compose Server. You must run this command before running any other command:

Syntax:
ComposeCli.exe connect [--url connection-url]
where:

url is the connection URL to the system where the server is running, such as https://machine.domain/attunitycompose. This is only required if the server is running on a remote machine.

Example:
ComposeCli.exe connect --url https://mymachine.mydomain/attunitycompose
Compose Control Program started...
Compose Control Program completed successfully.

Exporting a Project
You can use the Compose CLI to export a project.
Exported projects includes the following:

- Databases
- Model definitions (entities and attributes)
Mappings
Custom ETLs
Data warehouse ETL tasks
Data mart definitions

**Note** Existing data warehouse tables and generated ETL sets are not exported. Notifications and schedules are also not exported as they are considered to be environment-specific.

Syntax:

```
ComposeCli.exe export_project_repository --project project_name --outfile output-file [--is_without_credentials] [--password password] [--master_user_password master_user_password]
```

where:
- **project** is the name of the project you want to export.
- **outfile** specifies the path to and name of the output file. This file is in JSON format. For example: C:\file.json.
- **is_without_credentials** specifies to export the project settings without the encrypted fields. When importing to a new project, you will need to manually enter the project passwords (in the Compose database connection settings) after the import completes. In addition to eliminating the need to specify a password when exporting or importing the project, the **is_without_credentials** parameter also allows the project to be used in every Compose installation, regardless of its master user password. It is also useful in the event that you would like to keep the existing passwords in the target environment (e.g. when exporting from a testing environment to an existing project in the production environment).
- **password** specifies the password for encrypting the credentials in the exported project. The **password** qualifier must be used together with the **master_user_password** qualifier described below. Use the **password** qualifier if you want to encrypt the credentials in the exported project, but do not want the source master password to be used in a different environment. The specified password must be at least 32 characters in length and can either be user-devised or generated using the **genpassword** utility described in Changing the Master User Password.
- **master_user_password** the master user password defined for the source machine. This must be used together with the **password** qualifier. Use the **master_user_password** qualifier if you want to encrypt the credentials in the exported project, but do not want the source master password to be used in a different environment. In such a case, when you import the project to an environment that has a different master password, you will only need to specify the **password** qualifier.

For instructions on changing the master user password, see Changing the Master User Password.
Importing a Project
You can use the Compose CLI to import a project. If you import to an existing project, all of the project settings, except the project configuration items will be overridden. For information on the project configuration items, see Exporting the Project Configuration.

Imported projects includes the following:
» Databases
» Model definitions (entities and attributes)
» Mappings
» Custom ETLs
» Data warehouse ETL tasks
» Data mart definitions

Syntax:
ComposeCli.exe import_project_repository --project project_name --infile input-file [--password password] [--is_without_credentials] [--override_configuration] [--dont_backup_existing_project]

where:
» project is the name of the project you want to import.
» infile specifies the full path to the input file (including the file name). This file is in JSON format. For example: C:\file.json
» override_configuration overrides the existing project configuration. When importing a project, the default is not to override the existing project configuration.
» dont_backup_existing_project specifies not to back up the existing project. By default, existing projects are backed up to the following location (and automatically restored if the import fails):
<product_dir>\data\projects\<project_name>_backup_<timestamp>
» is_without_credentials specifies to import the project settings without the encrypted fields. In this case, you will need to manually enter the project passwords (in the Compose database connection settings).
» password the password specified with the password qualifier during export.

For instructions on changing the master user password, see Changing the Master User Password.
Note: Existing data warehouse tables and generated ETL sets are not imported. After the import completes, you must perform steps 3-4 below. You may also need to perform step 1 or 2, depending on whether you changed the data warehouse connection settings (step 1) or kept the existing connection settings (step 2).

1. If you changed the data warehouse connection settings after importing the project, then you need to create the tables in the new data warehouse.
2. If you edited the model in a testing environment and then imported the project into a production environment, then you need to validate and adjust the data warehouse.
3. Generate the data warehouse ETL set.
4. Generate the data mart ETL set.

For information on validating the data warehouse and generating the ETL set, see Creating and Managing the Data Warehouse. For information on generating the data mart ETL set, see Creating and Managing Data Marts.

See also: Moving Projects from the Test Environment to the Production Environment and Import/Export Scenarios - When is a Password Required?

Exporting the Project Configuration
You can use the Compose CLI to export the configuration settings of an existing project. This includes database definitions, scheduling jobs, and notifications. This is helpful, for example, when you need to migrate configuration settings from a test environment to the production environment.

For information about migrating projects, see Moving Projects from the Test Environment to the Production Environment.

Syntax:

ComposeCli.exe export_project_repository_config --project project_name --outfile output_file [--is_without_credentials] [--password password] [--master_user_password master_user_password]

where:

» project is the name of the project you want to export.
» outfile specifies the path to and name of the output file. This file is in JSON format. For example: C:\file.json.
» is_without_credentials specifies to export the project configuration without the encrypted fields. When importing to a new project, you will need to manually enter the project passwords (in the Compose database connection settings) after the import completes. In addition to eliminating the need to specify a password when exporting or importing the project, the is_without_credentials parameter also allows the project configuration to be used in every Compose installation, regardless of its Master User Password. It is also useful in the event that you would like to keep the existing pass-
words in the target environment (e.g. when exporting from a testing environment to an existing project in the production environment).

- `password` specifies the password for encrypting the credentials in the exported project configuration. The `password` qualifier must be used together with the `master_user_password` qualifier described below. Use the `password` qualifier if you want to encrypt the credentials in the exported project configuration, but do not want the source master password to be used in a different environment. The specified password must be at least 32 characters in length and can either be user-devised or generated using the `gen-password` utility described in Changing the Master User Password.

- `master_user_password` the master user password defined for the source machine. This must be used together with the `password` qualifier. Use the `master_user_password` qualifier if you want to encrypt the credentials in the exported project configuration, but do not want the source Master User Password to be used in a different environment. In such a case, when you import the project configuration to an environment that has a different Master User Password, you will only need to specify the `password` qualifier. For instructions on changing the Master User Password, see Changing the Master User Password.

See also: Moving Projects from the Test Environment to the Production Environment and Import/Export Scenarios - When is a Password Required?

### Importing the Project Configuration

You can use the Compose CLI to import the configuration settings of an existing project. This includes database definitions, scheduling jobs, and notifications. This is helpful, for example, when you need to migrate configuration settings from a test environment to the production environment. For information about migrating projects, see Moving Projects from the Test Environment to the Production Environment.

**Note** Before you can import the project configuration, you must first run the `import_project_repository` command described in Importing a Project.

**Syntax:**

```
ComposeCli.exe import_project_repository_config --project project_name --infile input-file [--password password] [--is_without_credentials]
```

where:

- `project` is the name of the project you want to export.
- `infile` specifies the path to and name of the input file. This file is in JSON format. For example: C:\file.json.
- `is_without_credentials` specifies to import the project configuration without the encrypted fields. In this case, you will need to manually enter the project configuration passwords (in the database connection settings).
- `password` the password specified with the `password` qualifier when the project
configuration was exported.
For instructions on changing the Master User Password, see Changing the Master User Password.
See also: Moving Projects from the Test Environment to the Production Environment and Import/Export Scenarios - When is a Password Required?

**Moving Projects from the Test Environment to the Production Environment**

After successfully creating and testing projects in the test environment, you now want to move those projects to the production environment. You also need to propagate updates from the testing environment to the production environment as necessary. Although it sounds complicated, moving new and updated projects from the test environment to the production environment is actually quite straightforward, as explained below.

See also Import/Export Scenarios - When is a Password Required?.

**Note**  The data source and data warehouse display names must be identical in both the testing and the production environments.

**To perform the initial migration from the testing environment to the production environment:**

1. Export the project from the test environment as described in Exporting a Project.
2. Import the test project to the production environment as described in Importing a Project.
3. Edit the connection settings to point to the production data source and data warehouse.
   
   For more information, see Setting up Data Source Connections and Setting up a Data Warehouse Connection respectively.
4. Configure notifications and scheduling as needed.
   
   For more information, see Scheduling Tasks and Defining Notifications Rules respectively.

**To propagate updates from the testing environment to the production environment:**

1. Export the project from the test environment as described in Exporting a Project.
2. Import the test project to the production environment as described in Importing a Project.

**Import/Export Scenarios - When is a Password Required?**

The following section describes which of the various export/import scenarios require a password to be specified.
Scenario 1: Moving a project or project configuration between two Compose machines without retaining the project credentials. This is useful when importing to a new project that will have different project credentials.

In such a scenario, simply add the `is_without_credentials` parameter to either the export or the import command.

Scenario 2: Moving a project or project configuration between two Compose machines that have the same Master User Password.

In such a scenario, neither the export command nor the import command need to include a password. If you do not want the source and target projects to have the same credentials (for database connectivity, etc.), then you also need to specify the `is_without_credentials` parameter in either the export or the import command.

Scenario 3: Moving a project or project configuration between two Compose machines that have a different Master User Password, but without revealing the Master User Password of the source machine.

In such a scenario, the export command must include the `password` and `master_user_password` parameters while the import command must include the `password` parameter. The same password (specified with the `password` parameter) must be used for both export and import.

Scenario 4: Moving a project or project configuration between two Compose machines that have a different Master User Password.

In such a scenario, the export command does not need to include a password, but the import command should specify the Master User Password of the source machine (using the `password` parameter).

Exporting Project Documentation

You can export a project to a zip file for record keeping and sharing offline. The project is exported as HTML files which can be easily printed to PDF using the print toolbar button in the HTML page.

To export the project documentation:
1. Open the project as described in Managing Projects.
2. Click the downward arrow to the right of the project name and select Generate Project Documentation from the drop-down menu.

   A zip file with the name of the project and a timestamp of when the documentation was generated will be created (e.g. MyComposeProject_documentation_03_22_2016_15_01_10.zip). Depending on your browser settings, the file will either be
automatically downloaded to your browser’s Downloads folder or you will be prompted to save it.

3. To view the documentation, extract the contents of the zip file and then open the index.html file.

A browser tab will open displaying the documentation categories in the left pane.

4. Navigate through the documentation using the tree in the left pane and the breadcrumbs above the documentation.

Viewing and Downloading DDL Scripts

In the DDL Script Files dialog box, you can view and download the data warehouse DDL script files. By default, Compose executes the Create, Adjust and Drop statements immediately upon user request. However, when the Generate DDL scripts but do not run them option is enabled, Compose will only generate the scripts but not execute them.

For more information on the Create DDL scripts only option, see Editing the Project Settings.

To open the DDL Script Files dialog box:

1. Open your project as described in Managing Projects.
2. Click the downward arrow to the right of the project name and select Show DDL Scripts from the drop-down menu.

The DDL Script Files dialog box opens.
3. To view a script, select the desired script in the Script Files pane on the left. The script will be displayed on the right.

4. To download a script, select the desired script in the Script Files pane on the left. Then click the download button in the top right of the dialog box.

5. To search for an element in the script, start to type in the search box. All strings that match the search query will be highlighted blue.
   You can navigate between search query matches using the arrows to the right of the search box. Use the right and left single arrows to navigate matches sequentially. Use the right and left double arrows to jump to the last and first match respectively.

6. To reset the search, either delete the search query or click the "x" to the right of the search box.

Project Versioning
Compose provides built-in project version control using the Git engine. Version control enables Compose developers to commit project revisions to both a local and a remote Git repository. If a mistake is made, Compose developers can easily roll back to earlier versions of the project while minimizing disruption to all team members.

Note  Revisions only store model and mapping information. After you revert to a saved revision, you will need to recreate the data warehouse and data mart tables.
To define Version Control Settings
1. From the project drop-down menu, select Version Control > Settings.
   The Version Control Settings - Git window opens.
   The Local Commits area shows the local root folder where project revisions are committed. The first time a project revision is committed, Compose creates a JSON file with the current project settings. The <project_name>.json file is archived to a ZIP file (<project_name>_deployment.zip), which is located in a project-specific folder under the source-control folder.
2. To enable commits to a remote Git database, select Enable remote commits and then provide the following information:
   » URL - The address of the remote Git database.
   » User name - Your user name for accessing the remote Git database.
   » Password - Your password for accessing the remote Git database.

To commit a project to Version Control
1. From the project drop-down menu, select Version Control > Commit.
   The Commit - <Project_Name> window opens.
   Enter a message in the Message box and optionally select the Remote push check box. Note that the Remote push check box will be disabled if the Enable remote commits option described above is not selected.
2. To enable commits to a remote Git database, select Enable remote commits and then provide the following information:
   » URL - The address of the remote Git database.
   » User name - Your user name for accessing the remote Git database.
   » Password - Your password for accessing the remote Git database.
To revert to a saved revision

1. From the project drop-down menu, select Version Control > Revisions history.
   The Revision History- <Project_Name> window opens.
   By default, the last 10 revisions are shown. You can change this number by selecting one of the available options from the Show drop-down list.
2. Optionally, use the Search box to find a specific revision.
3. Select the desired revision and then click the Deploy to Revision toolbar button.
4. When prompted to confirm the operation, click Yes.
   The existing project will be replaced.
5. Click Close to close the Revision History- <Project_Name> window.

To download a saved revision

1. From the project drop-down menu, select Version Control > Revisions history.
   The Revision History- <Project_Name> window opens.
   By default, the last 10 revisions are shown. You can change this number by selecting one of the available options from the Show drop-down list.
2. Optionally, use the Search box to find a specific revision.
3. Select the desired revision and then click the Download Revision as Package toolbar button.
   The package will be saved as a ZIP file in your browser's default download location.
This section explains how to set up data warehouse connectivity in an Attunity Compose project.

**In this chapter:**
- Adding a Data Warehouse in Compose
- Using Oracle as a Data Warehouse
- Using Microsoft SQL Server as a Data Warehouse
- Using Teradata as a Data Warehouse
- Using Amazon Redshift as a Data Warehouse
- Managing Databases

### Adding a Data Warehouse in Compose

The data warehouse contains the landing area tables (the target of the Attunity Replicate task), the staging tables, the actual data warehouse tables and the data mart tables.

For more information on the data warehouse structure in an project Attunity Compose project, see [Attunity Compose Architecture](#).

Note that Attunity Compose will not let you add data sources before you add a data warehouse. This is because the server connection settings for the source landing area are derived from the data warehouse settings.

For more information on adding data sources, see [Setting up Data Source Connections](#).

**To add a data warehouse:**

1. Open your project and click **Manage** in the bottom left of the **Databases** panel.
   - The **Manage Databases** dialog box opens.
2. Click the **New** toolbar button.
   - The **New Data Warehouse** dialog box opens.
3. Specify a display name for your database.
4. From the **Type** drop-down list, select the desired data warehouse.
5. Continue from one of the following according to the data warehouse you selected:
   - Using Oracle as a Data Warehouse
   - Using Microsoft SQL Server as a Data Warehouse
Using Oracle as a Data Warehouse

This section describes how to set up Oracle as a data warehouse in a Compose project.

**Note** When loading a huge number of records (i.e. hundreds of millions), the UNDO/REDO/TEMP tablespace on the Oracle database must be large enough to hold the data being loaded.

It contains the following topics:

- Prerequisites
- Data Types
- Defining the Connection Parameters

Prerequisites

The following sections list the prerequisites for using Oracle as a data warehouse in an Attunity Compose project.

**Client**

Before you can use Oracle as a source in an Attunity Compose project, make sure that the following client prerequisites have been met:

- The Oracle database should be configured with the required permissions (see below) and accessible from the Compose machine.
- Install Oracle Data Access Components (x64) on the computer where Attunity Compose is located. Then, add the full path of the Oracle Data Access DLL to the system environment variables.

The default path should be:

```
<ORACLE_PRODUCT_CLIENT_DIR>\ODP.NET\bin\4\n
```

**Note** The Attunity Compose service needs to be restarted after installing the required components.

- Install Oracle Instant Client for Microsoft Windows (x64) on the computer where Attunity Compose is located.
- If you want to use an Oracle TNS name in the connection settings, you first need to set the ORACLE_HOME environment variable.

**Example:**

```
<ORACLE_PRODUCT_CLIENT_DIR>\product\12.1.0\client_1
```
Permissions

To use Oracle as a Data Warehouse an Attunity Compose project, the Compose user must be granted the following privileges in the Oracle database:

» SELECT ANY TRANSACTION
» SELECT on V$NLS_PARAMETERS
» SELECT on V$TIMEZONE_NAMES
» SELECT on ALL_INDEXES
» SELECT on ALL_OBJECTS
» SELECT on DBA_OBJECTS
» 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
» DROP ANY TABLE
» SELECT ANY TABLE
» INSERT ANY TABLE
» INSERT ANY TABLE
» UPDATE ANY TABLE
» CREATE ANY VIEW
» DROP ANY VIEW
» CREATE ANY PROCEDURE
» ALTER ANY PROCEDURE
» DROP ANY PROCEDURE
» CREATE ANY SEQUENCE
» ALTER ANY SEQUENCE
» DROP ANY SEQUENCE

You can add the following permissions to use a specific table list:

» SELECT on <any-replicated-table>
» ALTER on <any-replicated-table>

The following permission must be granted for logon:

» CREATE SESSION

The Attunity Compose user must also be granted read permissions for the following DBA tables:

» SELECT on DBA_USERS
» SELECT on DBA_TAB_PRIVS
SELECT on DBA_OBJECTS
SELECT on DBA_SYNONYMS
SELECT on DBA_SEQUENCES
SELECT on DBA_TYPES
SELECT on DBA_INDEXES
SELECT on DBA_TABLES
SELECT on DBA_TRIGGERS

Note: If any of the required privileges cannot be granted to a V$xxx, then grant them to the V_$xxx.

Data Types
The Oracle database for Attunity Compose supports most Oracle data types. The following table shows the Oracle data warehouse data types that are supported when using Attunity Compose and the default mapping from Attunity Compose 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.

**Table 5.1 | Supported Oracle Data Types with Mapping from Attunity Compose Data Types**

<table>
<thead>
<tr>
<th>Attunity Compose Data Types</th>
<th>Oracle Data Types</th>
</tr>
</thead>
<tbody>
<tr>
<td>Varchar</td>
<td>VARCHAR (${LENGTH})</td>
</tr>
<tr>
<td>NVarchar</td>
<td>VARCHAR (${LENGTH})</td>
</tr>
<tr>
<td>Bigint</td>
<td>DECIMAL (19,0)</td>
</tr>
<tr>
<td>Date</td>
<td>DATE</td>
</tr>
<tr>
<td>Datetime</td>
<td>DATE</td>
</tr>
<tr>
<td>Decimal</td>
<td>DECIMAL (p,s)</td>
</tr>
<tr>
<td>Integer</td>
<td>DECIMAL (10,0)</td>
</tr>
<tr>
<td>GUID</td>
<td>VARCHAR (38)</td>
</tr>
<tr>
<td>IntAutoInc</td>
<td>DECIMAL (10,0)</td>
</tr>
<tr>
<td>Clob</td>
<td>CLOB</td>
</tr>
<tr>
<td>NClob</td>
<td>NCLOB</td>
</tr>
<tr>
<td>Byte</td>
<td>raw (${LENGTH})</td>
</tr>
</tbody>
</table>
Defining the Connection Parameters
This section describes how to use an Oracle database as a data warehouse in an Attunity Compose project.

To define Oracle as a data warehouse:
1. Open the New Data Warehouse dialog box as described in Adding a Data Warehouse in Compose.
2. Enter the information as described in the table below.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Specify a display name</td>
</tr>
<tr>
<td>Type</td>
<td>Select Oracle.</td>
</tr>
<tr>
<td>Server Name</td>
<td>Specify the name or IP address of the Oracle server machine.</td>
</tr>
<tr>
<td>Port</td>
<td>If you specified a TNS name in the Server Name field, make sure that this field is empty. Optionally, change the default port.</td>
</tr>
<tr>
<td>User Name</td>
<td>Specify your username for accessing the Oracle database. The specified user must have read/write privileges on the Oracle database.</td>
</tr>
<tr>
<td>Password</td>
<td>Specify your password for accessing the Oracle database.</td>
</tr>
<tr>
<td>SID</td>
<td>If you specified a TNS name in the Server Name field, make sure that this field is empty. Otherwise, specify the Oracle SID.</td>
</tr>
<tr>
<td>Data Warehouse Schema</td>
<td>Specify the schema in which to create the data warehouse tables.</td>
</tr>
<tr>
<td>Data Mart Schema</td>
<td>Specify the schema in which to create the data mart.</td>
</tr>
<tr>
<td>Maximum length of identifier names is 128</td>
<td>If your Oracle version is 12.2 or above and you are not working in compatibility mode, select this option to allow column and table names up to 128 characters.</td>
</tr>
</tbody>
</table>

Note: This option is only available for new projects. If you upgrade your Oracle version to 12.2 or above and it is being used in an existing project, please contact Attunity Support if you wish to use this option.

3. Click Test Connection to verify that Compose is able to establish a connection with the specified data warehouse.
4. Click OK to save your settings.
The database is added to the list on the left side of the Manage Endpoint Connections dialog box.

Using Microsoft SQL Server as a Data Warehouse
This section describes how to set up Microsoft SQL Server as a data warehouse in a Compose project.

It contains the following topics:

» Prerequisites
» Data Types
» Defining the Connection Parameters

Prerequisites
The following topic lists the prerequisites for using Microsoft SQL Server as a data warehouse in an Attunity Compose project.

Client
Microsoft SQL Server Native Client must be installed on the Attunity Compose machine.

Permissions
To use Microsoft SQL Server as a Data Warehouse an Attunity Compose project, the Compose user must be granted the following privileges in the Microsoft SQL Server database:

» The Attunity Compose user must have at least the db_owner user role on the Microsoft SQL Server database.
» The Attunity Compose user must be granted the CREATE VIEW permission on the Microsoft SQL Server database.
» A Microsoft SQL Server system administrator must provide this permission for all Attunity Compose users.

Working with Windows Authentication
You can configure the Attunity Compose Microsoft SQL Server source to log in to Microsoft SQL Server 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 Compose user is specified as the "Log on as" user for the Attunity Compose Server service account.

OR
Microsoft SQL Server is configured to allow login for the Attunity Compose Server service account.

Data Types
The following table shows the Microsoft SQL Server data warehouse data types that are supported when using Attunity Compose and the default mapping from Attunity Compose 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.

<table>
<thead>
<tr>
<th>Attunity Compose Data Types</th>
<th>Microsoft SQL Server Data Types</th>
</tr>
</thead>
<tbody>
<tr>
<td>NVarchart</td>
<td>NVARCHAR (Length)</td>
</tr>
<tr>
<td>Varchart</td>
<td>VARCHAR (Length)</td>
</tr>
<tr>
<td>Bigint</td>
<td>BIGINT</td>
</tr>
<tr>
<td>Decimal</td>
<td>NUMERIC (p,s)</td>
</tr>
<tr>
<td>Integer</td>
<td>INT</td>
</tr>
<tr>
<td>Date</td>
<td>DATETIME2</td>
</tr>
<tr>
<td>Datetime</td>
<td>DATETIME2</td>
</tr>
<tr>
<td>GUID</td>
<td>UNIQUEIDENTIFIER</td>
</tr>
<tr>
<td>IntAutoInc</td>
<td>INT IDENTITY</td>
</tr>
<tr>
<td>Clob</td>
<td>VARCHAR (MAX)</td>
</tr>
<tr>
<td>NClob</td>
<td>NVARCHAR (MAX)</td>
</tr>
<tr>
<td>Byte</td>
<td>VARBINARY (Length)</td>
</tr>
</tbody>
</table>

Defining the Connection Parameters
This section describes how to use a Microsoft SQL Server database as a data warehouse in an Attunity Compose project.

To define Microsoft SQL Server as a data warehouse:
1. Open the New Data Warehouse dialog box as described in Adding a Data Warehouse in Compose.
2. Enter the information as described in the table below.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Specify a display name</td>
</tr>
<tr>
<td>Type</td>
<td>Select <strong>Microsoft SQL Server</strong>.</td>
</tr>
<tr>
<td>Server Name</td>
<td>Specify the name or IP address of the Microsoft SQL Server server machine.</td>
</tr>
<tr>
<td>Port</td>
<td>Optionally, change the default port.</td>
</tr>
<tr>
<td>Windows authentication</td>
<td>Choose how you want Compose to log in to the Microsoft SQL Server database. If you choose <strong>Windows authentication</strong>, see <em>Working with Windows Authentication</em> below.</td>
</tr>
<tr>
<td>SQL Server authentication</td>
<td></td>
</tr>
<tr>
<td>User Name</td>
<td>Specify your user name for accessing the Microsoft SQL Server database. The specified user must have read/write privileges on the Microsoft SQL Server database.</td>
</tr>
<tr>
<td>Password</td>
<td>Specify your password for accessing the Microsoft SQL Server database.</td>
</tr>
<tr>
<td>Database Name</td>
<td>Specify the name of the Microsoft SQL Server database.</td>
</tr>
<tr>
<td>Data Warehouse Schema</td>
<td>Specify the schema in which to create the data warehouse tables.</td>
</tr>
<tr>
<td>Data Mart Schema</td>
<td>Specify the schema in which to create the data mart.</td>
</tr>
</tbody>
</table>

3. Click **Test Connection** to verify that Compose is able to establish a connection with the specified data warehouse.
4. Click **OK** to save your settings.
   The database is added to the list on the left side of the **Manage Endpoint Connections** dialog box.

**Working with Windows Authentication**
You can configure Attunity Compose to log in to Microsoft SQL Server 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 Attunity Compose user is specified as the "Log on as" user for the Attunity Compose service account.
-OR-
Microsoft SQL Server is configured to allow login for the Attunity Compose service account.

Using Teradata as a Data Warehouse
This section describes how to set up Teradata as a data warehouse in a Compose project.

It contains the following topics:
- Prerequisites
- Data Types
- Defining the Connection Parameters

Prerequisites
The following topic lists the prerequisites for using Teradata as a data warehouse in an Attunity Compose project.

Client
Before you can use Teradata as a data warehouse in an Attunity Compose project, make sure that the following client prerequisites have been met:
- The Teradata database should be configured with the required permissions (see below) and accessible from the Compose machine.
- Install Teradata Database ODBC Driver on the Compose machine
- Install .NET provider for Teradata version 15.11 on the Compose machine
- Install Teradata JDBC Driver on the Compose machine and copy the `tdgssconfig.jar` and `terajdbc4.jar` jar files to:
  `<Compose_Installation_Folder>\java\jdbc`

Permissions
To use Teradata as a Data Warehouse an Attunity Compose project, the Compose user must be granted the following privileges in the Teradata database:
- 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>

Data Types
The following table shows the Teradata data warehouse data types that are supported when using Attunity Compose and the default mapping from Attunity Compose 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.

**Table 5.3 | Supported Teradata Data Types with Mapping from Attunity Compose Data Types**

<table>
<thead>
<tr>
<th>Attunity Compose Data Types</th>
<th>Teradata Data Types</th>
</tr>
</thead>
<tbody>
<tr>
<td>NVARCHAR</td>
<td>VARCHAR (Length)</td>
</tr>
<tr>
<td>VARCHAR</td>
<td>VARCHAR (Length)</td>
</tr>
<tr>
<td>BIGINT</td>
<td>BIGINT</td>
</tr>
<tr>
<td>DECIMAL</td>
<td>DECIMAL (p,s)</td>
</tr>
<tr>
<td>INTEGER</td>
<td>INTEGER</td>
</tr>
<tr>
<td>DATE</td>
<td>TIMESTAMP (6)</td>
</tr>
<tr>
<td>DATETIME</td>
<td>TIMESTAMP (6)</td>
</tr>
<tr>
<td>GUID</td>
<td>VARCHAR (38)</td>
</tr>
<tr>
<td>INTAUTOINC</td>
<td>integer (GENERATED ALWAYS AS IDENTITY)</td>
</tr>
<tr>
<td>CLOB</td>
<td>CLOB</td>
</tr>
<tr>
<td>NCLOB</td>
<td>CLOB</td>
</tr>
<tr>
<td>BYTE</td>
<td>BYTEINT</td>
</tr>
</tbody>
</table>

Defining the Connection Parameters
This section describes how to use an Teradata database as a data warehouse in an Attunity Compose project.

**To define Teradata as a data warehouse:**
1. Open the New Data Warehouse dialog box as described in Adding a Data Warehouse in Compose.
2. Enter the information as described in the table below.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Specify a display name</td>
</tr>
<tr>
<td>Type</td>
<td>Select <strong>Teradata</strong>.</td>
</tr>
<tr>
<td>Port</td>
<td>Optionally, change the default port.</td>
</tr>
<tr>
<td>Server Name</td>
<td>Specify the name or IP address of the Teradata server machine.</td>
</tr>
<tr>
<td>User Name</td>
<td>Specify your username for accessing the Teradata database. The specified user must have read/write privileges on the Teradata database.</td>
</tr>
<tr>
<td>Password</td>
<td>Specify your password for accessing the Teradata database.</td>
</tr>
<tr>
<td>Database Name</td>
<td>Specify the database in which to create the data warehouse tables.</td>
</tr>
<tr>
<td>Data Mart Database</td>
<td>Specify the database in which to create the data mart.</td>
</tr>
<tr>
<td>Authentication Method</td>
<td>The expected authentication method is set in an XML file by the TeraGSS program, tdgssconfigure. The following authentication methods are available:</td>
</tr>
<tr>
<td></td>
<td>» Empty - The same as omitting the keyword.</td>
</tr>
<tr>
<td></td>
<td>» <strong>TD1</strong> - Selects Teradata 1 as the authentication mechanism. User name and password are required.</td>
</tr>
<tr>
<td></td>
<td>» <strong>TD2</strong> - Selects Teradata 2 as the authentication mechanism. User name and password are required.</td>
</tr>
<tr>
<td></td>
<td>» <strong>TDNEGO</strong> - Selects one of the Authentication Mechanisms automatically based on the policy, without user involvement.</td>
</tr>
<tr>
<td></td>
<td>» <strong>LDAP</strong> - Selects Lightweight Directory Access Protocol (LDAP) as the Authentication Mechanism. The application provides the user name and password.</td>
</tr>
<tr>
<td></td>
<td>» <strong>KRB5</strong> - Selects Kerberos (KRB5) on Windows clients working with Windows servers. The application provides the user name and password.</td>
</tr>
<tr>
<td></td>
<td>» <strong>NTLM</strong> - Selects NT LAN Manager (NTLM) on Windows clients working with Windows servers. The application provides the user name and password.</td>
</tr>
<tr>
<td>Additional JDBC</td>
<td>Additional parameters to add to the default JDBC connection string.</td>
</tr>
<tr>
<td>Parameters</td>
<td>These should be key values separated by a semi-colon.</td>
</tr>
<tr>
<td></td>
<td><strong>Example:</strong> LOGMECH=LDAP;KEY=VALUE;KEY1=VALUE1</td>
</tr>
</tbody>
</table>
### Field Description

<table>
<thead>
<tr>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>By default, Compose connects to Teradata in ANSI mode. To run a Teradata stored procedure that was created in TERA (Teradata) mode, you need to add the parameter TMODE=TERA to the Additional JDBC Parameters field. For information about creating a custom ETL and running it as a stored procedure, see Creating Custom ETL Scripts.</td>
</tr>
</tbody>
</table>

3. Click **Test Connection** to verify that Compose is able to establish a connection with the specified data warehouse.

4. Click **OK** to save your settings.
   - The database is added to the list on the left side of the Manage Endpoint Connections dialog box.

---

### Using Amazon Redshift as a Data Warehouse

This section describes how to set up Amazon Redshift as a data warehouse in a Compose project.

It contains the following topics:

- **Prerequisites**
- **Data Types**
- **Defining the Connection Parameters**

#### Prerequisites

For a list of prerequisites, see **PART 1: Satisfy the Prerequisites** of the Attunity Compose for Amazon Redshift Tutorial. To use Amazon Redshift as a data warehouse in your Compose project, you also need to **purchase the Attunity Compose for Amazon Redshift AMI** from Amazon Marketplace.

Note that you will also need to meet the prerequisites for the specific source endpoint defined in your Attunity Replicate task (e.g. Oracle). For a list of these prerequisites, refer to the **Attunity Replicate User Guide and Reference**.

It is also highly recommended to perform the **Attunity Compose for Amazon Redshift Tutorial** before deploying in a production environment.
Permissions

To use Amazon Redshift as a Data Warehouse an Attunity Compose project, the Compose user must be granted the following privileges in the Amazon Redshift database:

- SELECT, INSERT, UPDATE and DELETE
- Bulk Load
- CREATE, ALTER, DROP
- CREATE VIEW

Unless the user is the DB Owner, the user specified in the Amazon Redshift connection settings must be granted the above permissions.

Setting up SSL

The Attunity Compose for Amazon Redshift AMI comes preinstalled with the Amazon Redshift public certificate, which it uses to establish a secure connection (SSL) to the Amazon Redshift Cluster. Since Compose always connects to Amazon Redshift using SSL, you need to replace the public certificate before it expires.

To check the certificate's expiration date

- Run the following command
  
  C:\Program Files\Java\jre<version>\bin>keytool -printcert -file "C:\Program Files\Attunity\Compose\java\bin\redshift-ssl-ca-cert.pem"

  The expiration date will be displayed in the "until" field:

  ![Certificate Details]

To replace the certificate

1. Download the Amazon Redshift public key from here:
   https://s3.amazonaws.com/redshift-downloads/redshift-ssl-ca-cert.pem

2. Make a copy of the file and rename it to root.crt. Then, copy the root.crt file to the following directory:
   C:\Windows\System32\config\systemprofile\AppData\Roaming\postgresql\ 

3. Copy the original file (redshift-ssl-ca-cert.pem) to the following directory:
   <product_dir>\java\bin\ 

4. Open a command prompt as administrator and run the following command to register the certificate:
   <product_dir>\bin\acjs.bat server importcert redshift-ssl-ca-cert.pem redshift

   The following message should be displayed:
   Status: 0 - Certificate imported
Data Types
The following table shows the Amazon Redshift data warehouse data types that are supported when using Attunity Compose and the default mapping from Attunity Compose 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.

<table>
<thead>
<tr>
<th>Attunity Compose Data Types</th>
<th>Amazon Redshift Data Types</th>
</tr>
</thead>
<tbody>
<tr>
<td>INTEGER</td>
<td>INT4</td>
</tr>
<tr>
<td>BIGINT</td>
<td>INT8</td>
</tr>
<tr>
<td>DECIMAL</td>
<td>NUMERIC (p,s)</td>
</tr>
<tr>
<td>DATE</td>
<td>DATE</td>
</tr>
<tr>
<td>VARCHAR</td>
<td>VARCHAR (Length)</td>
</tr>
<tr>
<td>GUID</td>
<td>VARCHAR (38)</td>
</tr>
<tr>
<td>DATETIME</td>
<td>TIMESTAMP</td>
</tr>
<tr>
<td>BYTE</td>
<td>VARCHAR (Length in Bytes)</td>
</tr>
<tr>
<td>TIME</td>
<td>VARCHAR (20)</td>
</tr>
</tbody>
</table>

Defining the Connection Parameters
This section describes how to use an Amazon Redshift database as a data warehouse in an Attunity Compose project.

To define Amazon Redshift as a data warehouse:
1. Open the New Data Warehouse dialog box as described in Adding a Data Warehouse in Compose.
2. Enter the information as described in the table below.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Specify a display name</td>
</tr>
<tr>
<td>Type</td>
<td>Select Amazon Redshift.</td>
</tr>
<tr>
<td>Server Name</td>
<td>Specify the name or IP address of the Amazon Redshift cluster.</td>
</tr>
</tbody>
</table>
### Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Port</td>
<td>Optionally, change the default port.</td>
</tr>
<tr>
<td>User Name</td>
<td>Specify your user name for accessing the Amazon Redshift database.</td>
</tr>
<tr>
<td></td>
<td>The specified user must have read/write privileges on the Amazon Redshift database.</td>
</tr>
<tr>
<td>Password</td>
<td>Specify your password for accessing the Amazon Redshift database.</td>
</tr>
<tr>
<td>Database Name</td>
<td>Specify the name of the Amazon Redshift database.</td>
</tr>
<tr>
<td>Data Warehouse Schema</td>
<td>Specify the schema in which to create the data warehouse tables.</td>
</tr>
<tr>
<td>Data Mart Schema</td>
<td>Specify the schema in which to create the data mart.</td>
</tr>
<tr>
<td>More Options</td>
<td>Click to see or hide advanced options.</td>
</tr>
</tbody>
</table>

**Character column size in bytes**

This should be calculated according to the largest value you are likely to store in a `VARCHAR` column. Tables in the Landing Area will be divided by the specified value (and rounded up). So, for example, if the value of **Character column size in bytes** is 3 (the default), both `VARCHAR (12 bytes)` and `VARCHAR (10 bytes)` will be discovered as `VARCHAR (4 characters).

**See also:**

**Note** If this value is changed, existing tables will not be affected (i.e. the change will only take effect if new columns are added to the model and the data warehouse tables are updated accordingly).

3. Click **Test Connection** to verify that Compose is able to establish a connection with the specified data warehouse.

4. Click **OK** to save your settings.

   The database is added to the list on the left side of the **Manage Endpoint Connections** dialog box.
Managing Databases
You can edit and delete databases as required. The table below describes the available options.

<table>
<thead>
<tr>
<th>To</th>
<th>Do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit a database</td>
<td>In the left side of the Manage Databases dialog box, select the database that you want to edit and then click the Edit toolbar button.</td>
</tr>
<tr>
<td>Delete a database</td>
<td>In the left side of the Manage Databases dialog box, select the database that you want to delete and then click the Delete toolbar button.</td>
</tr>
<tr>
<td></td>
<td>Click Yes when prompted to confirm the deletion.</td>
</tr>
</tbody>
</table>
6 | Setting up Data Source Connections

This section explains how to set up data source connectivity in an Attunity Compose project.

In this chapter:
- Data Source Permissions
- Adding Data Sources in Compose
- Using Oracle as a Source
- Using Microsoft SQL Server as a Source
- Using MySQL as a Source
- Using IBM DB2 as a Source
- Using Teradata as a Source
- Using Amazon Redshift as a Source
- Managing Databases

Data Source Permissions
This section lists the required permissions for the source landing area and the source database defined in an Attunity Replicate task.

» Source Landing Area Permissions
» Source Database Permissions

Source Landing Area Permissions
For proper operation, the source landing area database must be granted the following permissions:

» Read metadata
» Select from tables
» Create tables for error marts
» Insert to tables (error marts)

For information on the landing area, see Landing Area Settings.
Source Database Permissions
If defined in a Compose project, the source database in the Replicate task must be granted the following permissions:
  ➤ Read metadata (Columns, Primary Keys and Foreign Keys)
  ➤ Select from tables
For information on when you need to define the source database in the Replicate task, see Source Database Connection.

Adding Data Sources in Compose
In a Compose project, you can define any number of data sources. Defining multiple data sources is necessary if the data that you eventually want to be available in your data mart (s) is located in several data sources.

Note that before you can define a data source in Attunity Compose, you first need to define a data warehouse.

For more information on adding data sources, see Setting up a Data Warehouse Connection.

To add a data source:
1. Open your project and click Manage in the Databases panel.
   The Manage Databases dialog box opens.
2. Click the New toolbar button.
   The New Data Source dialog box opens.
   The dialog box is divided into the following areas:
   ➤ Landing Area Settings - These settings are required. These settings tell Compose where the replica source tables are located. Since the landing area is always located in the data warehouse and the data warehouse connection details are already known to Compose, you do not need to provide them again.
   ➤ Associate with Replicate Task - Select this to associate your Compose project with the related Replicate task. Replicate tasks replicate the relevant tables from the source database to the landing area in your data warehouse. Specifying the Replicate task name will enable you to monitor and control that task from within Compose. However, before you can specify a Replicate task name, you first need to define the connection settings to the Attunity Replicate Server machine.
   To do this, click the Replicate Server Settings link below the Associate with Replicate task field and then configure the settings as described in Monitoring and Controlling Replicate Tasks.
   ➤ Source Database Connection - Select this if you want to generate the model by discovering the source database in the Replicate task (as opposed to discovering the landing area or importing a model from ERwin).
For more information on the discovery process, see Generating the Model.

3. In the Name field, specify a display name for your database.

4. Continue from Defining the Connection Parameters in one of the following:
   - Using Oracle as a Source
   - Using Microsoft SQL Server as a Source
   - Using MySQL as a Source
   - Using IBM DB2 as a Source
   - Using Teradata as a Source
   - Using Amazon Redshift as a Source

Using Oracle as a Source

This section describes how to set up Oracle as a source database in a Compose project.

It contains the following topics:

- Prerequisites
- Data Types
- Defining the Connection Parameters

Prerequisites

Before you can use Oracle as a source in an Attunity Compose project, make sure that the following prerequisites have been met:

- The Oracle database should be configured with the required Data Source Permissions and accessible from the Compose machine.
- Install Oracle Data Access Components (x64) on the computer where Attunity Compose is located. Then, add the full path of the Oracle Data Access DLL to the system environment variables.

The default path should be:

<ORACLE_PRODUCT_CLIENT_DIR>\ODP.NET\bin\4\n
**Note**  The Attunity Compose service needs to be restarted after installing the required components.

- Install Oracle Instant Client for Microsoft Windows (x64) on the computer where Attunity Compose is located.
- If you want to use an Oracle TNS name in the connection settings, you first need to set the ORACLE_HOME environment variable.

**Example:**

<ORACLE_PRODUCT_CLIENT_DIR>\product\12.1.0\client_1
Data Types
The Oracle database for Attunity Compose supports most Oracle data types. The following table shows the Oracle source data types that are supported when using Attunity Compose and the default mapping to Attunity Compose data types.

For information on how to view the data type that is mapped in the data warehouse, see the section for the data warehouse database you are using.

<table>
<thead>
<tr>
<th>Oracle Data Types</th>
<th>ATTUNITY COMPOSE DATA TYPES</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHAR</td>
<td>VARCHAR</td>
</tr>
<tr>
<td>VARCHAR2(30)</td>
<td>VARCHAR(30)</td>
</tr>
<tr>
<td>NCHAR(40)</td>
<td>VARCHAR(80)</td>
</tr>
<tr>
<td>NVARCHAR2(60)</td>
<td>VARCHAR(120)</td>
</tr>
<tr>
<td>NUMBER</td>
<td>DECIMAL</td>
</tr>
<tr>
<td>FLOAT</td>
<td>DECIMAL(38,12)</td>
</tr>
<tr>
<td>REAL</td>
<td>DECIMAL(38,12)</td>
</tr>
<tr>
<td>DATE</td>
<td>DATE</td>
</tr>
<tr>
<td>TIMESTAMP(6)</td>
<td>DATE</td>
</tr>
<tr>
<td>TIMESTAMP(6) WITH LOCAL TIME ZONE</td>
<td>DATE</td>
</tr>
<tr>
<td>TIMESTAMP(6) WITH TIME ZONE</td>
<td>DATE</td>
</tr>
<tr>
<td>DOUBLE PRECISION</td>
<td>DECIMAL(38,12)</td>
</tr>
</tbody>
</table>

Non-Supported Data Types
Source Oracle tables with columns of the following Oracle data types are not supported and will be ignored.

- BLOB
- CLOB
- NCLOB
- BFILE
- BINARY_FLOAT
- BINARY_DOUBLE
- INTERVAL YEAR (2) TO MONTH
- INTERVAL DAY (6) TO SECOND (5)
- RAW
- ROWID
Defining the Connection Parameters
You can add an Oracle database to Attunity Compose to use as a source.

To add an Oracle source database to Attunity Compose:
1. Open the New Data Source dialog box and specify a name for it, and specify a name for it, as described in Adding Data Sources in Compose.
2. Enter the information as described in the table below.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Landing Area Settings</strong></td>
<td></td>
</tr>
<tr>
<td>Content Type</td>
<td>Choose whether the content in the landing area is <strong>Full Load, Change Processing</strong> or <strong>Full Load and Change Processing</strong> (according to the Attunity Replicate task definition). See also After applying changes below.</td>
</tr>
<tr>
<td>Designated By</td>
<td>This field is read-only since the Oracle landing area is always designated by <strong>Schema</strong>. For more information, see Defining an Attunity Replicate Task.</td>
</tr>
<tr>
<td>Schema Name</td>
<td>Specify the schema name. This must be the same as the schema defined in the Oracle target connection string in the Attunity Replicate task. For more information, see Defining an Attunity Replicate Task.</td>
</tr>
<tr>
<td>Error Mart Schema Name</td>
<td>Specify the schema where you want the data mart exception tables to be created. Data that is rejected by data quality rules will be copied to tables in the specified schema. For more information on error marts, see Defining and Managing Data Quality Rules.</td>
</tr>
<tr>
<td>After applying changes</td>
<td>If you selected <strong>Change Processing</strong> or <strong>Full Load and Change Processing</strong> as the Content Type, you can determine whether the Change Tables will be deleted or archived after the changes have been applied (to the data warehouse tables). If you select Archive the Change Tables, you also need to specify a Database name and Schema name in the designated fields.</td>
</tr>
</tbody>
</table>

Source Database Connection - Select this option if you want to generate the model by discovering the source database.

Type: Select **Oracle**.
### Using Microsoft SQL Server as a Source

This section describes how to set up Microsoft SQL Server as a source database in a Compose project.

It contains the following topics:

- **Prerequisites**
- **Data Types**
- **Defining the Connection Parameters**

### Prerequisites

Before you can use Microsoft SQL Server as a source in an Attunity Compose project, make sure that the following prerequisites have been met:

- Microsoft SQL Server should be configured with the required Data Source Permissions and accessible from the Compose machine.
- Microsoft SQL Server Native Client must be installed on the Attunity Compose machine.
- Attunity Compose supports the following Microsoft SQL Server editions.
Working with Windows Authentication

You can configure the Attunity Compose Microsoft SQL Server source to log in to Microsoft SQL Server 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 Compose user is specified as the "Log on as" user for the Attunity Compose Server service account.
  
  OR

  - Microsoft SQL Server is configured to allow login for the Attunity Compose Server service account.

Data Types

The Microsoft SQL Server database for Attunity Compose supports most Microsoft SQL Server data types. The following table shows the Microsoft SQL Server source data types that are supported when using Attunity Compose and the default mapping to Attunity Compose data types.

For information on how to view the data type that is mapped in the data warehouse, see the section for the data warehouse database you are using.

<table>
<thead>
<tr>
<th>Microsoft SQL Server Data Types</th>
<th>Attunity Compose Data Types</th>
</tr>
</thead>
<tbody>
<tr>
<td>CHAR</td>
<td>VARCHAR</td>
</tr>
<tr>
<td>VARCHAR</td>
<td>VARCHAR</td>
</tr>
<tr>
<td>NCHAR</td>
<td>VARCHAR</td>
</tr>
<tr>
<td>NVARCHAR</td>
<td>VARCHAR</td>
</tr>
<tr>
<td>BIT</td>
<td>INTEGER</td>
</tr>
<tr>
<td>TINYINT</td>
<td>INTEGER</td>
</tr>
<tr>
<td>SMALLINT</td>
<td>INTEGER</td>
</tr>
<tr>
<td>int</td>
<td>INTEGER</td>
</tr>
<tr>
<td>bigint</td>
<td>BIGINT</td>
</tr>
</tbody>
</table>
### Table 6.2 | Supported Microsoft SQL Server Data Types with Mapping to Attunity Compose Data Types

<table>
<thead>
<tr>
<th>Microsoft SQL Server Data Types</th>
<th>Attunity Compose Data Types</th>
</tr>
</thead>
<tbody>
<tr>
<td>DECIMAL</td>
<td>DECIMAL</td>
</tr>
<tr>
<td>NUMERIC</td>
<td>DECIMAL</td>
</tr>
<tr>
<td>SMALLMONEY</td>
<td>DECIMAL(11,4)</td>
</tr>
<tr>
<td>MONEY</td>
<td>DECIMAL(20,4)</td>
</tr>
<tr>
<td>FLOAT</td>
<td>DECIMAL(38,12)</td>
</tr>
<tr>
<td>REAL</td>
<td>DECIMAL(18,6)</td>
</tr>
<tr>
<td>DATETIME</td>
<td>DATE</td>
</tr>
<tr>
<td>DATETIME2</td>
<td>DATE</td>
</tr>
<tr>
<td>SMALLDATETIME</td>
<td>DATE</td>
</tr>
<tr>
<td>BINARY</td>
<td>BYTE</td>
</tr>
<tr>
<td>DATE</td>
<td>DATE</td>
</tr>
<tr>
<td>TIME</td>
<td>VARCHAR(16)</td>
</tr>
<tr>
<td>UNIQUEIDENTIFIER</td>
<td>GUID</td>
</tr>
</tbody>
</table>

### Non-Supported Data Types

Source Microsoft SQL Server tables with columns of the following Microsoft SQL Server data types are not supported and will be ignored:

- VARCHAR (MAX)
- TEXT
- NVARCHAR (MAX)
- NTEXT
- VARBINARY
- IMAGE
- DATETIMEOFFSET
- TIMESTAMP
- SQL_VARIANT
- XML

### Defining the Connection Parameters

You can add a Microsoft SQL Server database to Attunity Compose to use as a source.
To add a Microsoft SQL Server source database to Attunity Compose:

1. Open the New Data Source dialog box and specify a name for it, as described in Adding Data Sources in Compose.

2. Enter the information as described in the table below.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Landing Area Settings</strong></td>
<td></td>
</tr>
<tr>
<td>Content Type</td>
<td>Choose whether the content in the landing area is <strong>Full Load Only</strong> or <strong>Full Load and Change Tables</strong> (according to the Attunity Replicate task definition).</td>
</tr>
<tr>
<td></td>
<td>See also After applying changes below.</td>
</tr>
<tr>
<td>Designated By</td>
<td>Select whether the landing area is a <strong>Database</strong> or a <strong>Schema</strong>. This will depend on how the target endpoint was defined in the Attunity Replicate task.</td>
</tr>
<tr>
<td></td>
<td>For more information, see Defining an Attunity Replicate Task.</td>
</tr>
<tr>
<td>Database Name</td>
<td>If the landing area is designated by a <strong>Database</strong>, specify the database name. This must be the same as the Microsoft SQL Server target database defined in the Attunity Replicate task.</td>
</tr>
<tr>
<td></td>
<td>For more information, see Defining an Attunity Replicate Task.</td>
</tr>
<tr>
<td>Schema Name</td>
<td>Specify the schema name. This must be the same as the schema name specified in the Attunity Replicate task.</td>
</tr>
<tr>
<td></td>
<td>For more information, see Defining an Attunity Replicate Task.</td>
</tr>
<tr>
<td>Error Mart Schema Name</td>
<td>Specify the schema where you want the data mart exception tables to be created. Data that is rejected by data quality rules will be copied to tables in the specified schema.</td>
</tr>
<tr>
<td></td>
<td>For more information on error marts, see Defining and Managing Data Quality Rules.</td>
</tr>
<tr>
<td>After applying changes</td>
<td>If you selected <strong>Change Processing</strong> or <strong>Full Load and Change Processing</strong> as the <strong>Content Type</strong>, you can determine whether the Change Tables will be deleted or archived after the changes have been applied (to the data warehouse tables). If you select <strong>Archive the Change Tables</strong>, you also need to specify a <strong>Database name</strong> and <strong>Schema name</strong> in the designated fields.</td>
</tr>
</tbody>
</table>

**Source Database Connection** - Select this option if you want to generate the model by discovering the source database.

<table>
<thead>
<tr>
<th>Type</th>
<th>Select <strong>Microsoft SQL Server</strong>.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server Name</td>
<td>Specify the name or IP address of the Microsoft SQL Server machine.</td>
</tr>
</tbody>
</table>
### Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Port</td>
<td>Optionally, change the default port.</td>
</tr>
<tr>
<td>Windows authentication</td>
<td>Choose how you want Compose to log in to the Microsoft SQL Server database. If you choose <strong>Windows authentication</strong>, see Working with Windows Authentication below.</td>
</tr>
<tr>
<td>SQL Server authentication</td>
<td></td>
</tr>
<tr>
<td>User Name</td>
<td>Specify your user name for accessing the Microsoft SQL Server database. The specified user must have read/write privileges on the Microsoft SQL Server database.</td>
</tr>
<tr>
<td>Password</td>
<td>Specify your password for accessing the Microsoft SQL Server database.</td>
</tr>
<tr>
<td>Database Name</td>
<td>Specify the name of the Microsoft SQL Server database.</td>
</tr>
<tr>
<td>Schema</td>
<td>Specify the schema containing the source tables.</td>
</tr>
</tbody>
</table>

3. Click **Test Connection** to verify that Compose is able to establish a connection with the specified database and/or landing area.

4. Click **OK** to save your settings.
   The database is added to the list on the left side of the **Manage Databases** dialog box.

### Working with Windows Authentication

You can configure Attunity Compose to log in to Microsoft SQL Server 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 Attunity Compose user is specified as the “Log on as” user for the Attunity Compose service account.
  - OR -
  - Microsoft SQL Server is configured to allow login for the Attunity Compose service account.

### Using MySQL as a Source

This section describes how to set up MySQL as a source database in a Compose project.

It contains the following topics:

- **Prerequisites**
- **Data Types**
Defining the Connection Parameters

Prerequisites
Before you can use MySQL as a source in an Attunity Compose project, make sure that the following prerequisites have been met:
- The MySQL database should be configured with the required Data Source Permissions and accessible from the Compose machine.
  - The following MySQL editions are supported:
    - MySQL Community Edition
    - MySQL Standard Edition
    - MySQL Enterprise Edition
    - MySQL Cluster Carrier Grade Edition
  - MySQL ODBC 64-bit client must be installed on the same computer as Attunity Compose.

Cluster Prerequisites
To be able to discover clustered (NDB) tables, the following parameters must be configured in the MySQL my.ini (Windows) file.

### Table 6.3 | Required my.ini/my.cnf Parameters for Cluster Replication

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ndb_log_bin</td>
<td>Must be: ndb_log_bin=on</td>
</tr>
<tr>
<td></td>
<td>This ensures that changes in clustered tables will be logged to the binary log.</td>
</tr>
<tr>
<td>ndb_log_update_as_write</td>
<td>Must be: ndb_log_update_as_write=OFF</td>
</tr>
<tr>
<td></td>
<td>This prevents writing UPDATEs as INSERTs in the binary log.</td>
</tr>
<tr>
<td>ndb_log_updated_only</td>
<td>Must be: ndb_log_updated_only=OFF</td>
</tr>
<tr>
<td></td>
<td>Ensures that the binary log will contain the entire row and not just the changed columns.</td>
</tr>
</tbody>
</table>

Data Types
The MySQL database for Attunity Compose supports most MySQL data types. The following table shows the MySQL source data types that are supported when using Attunity Compose and the default mapping to Attunity Compose data types.

For information on how to view the data type that is mapped in the data warehouse, see the section for the data warehouse database you are using.
Table 6.4 | Supported MySQL Data Types with Mapping to Attunity Compose Data Types

<table>
<thead>
<tr>
<th>MYSQL DATA TYPES</th>
<th>Attunity Compose Data Types</th>
</tr>
</thead>
<tbody>
<tr>
<td>BIGINT</td>
<td>BIGINT</td>
</tr>
<tr>
<td>BINARY</td>
<td>BYTE</td>
</tr>
<tr>
<td>BIT</td>
<td>BIGINT</td>
</tr>
<tr>
<td>CHAR</td>
<td>VARCHAR</td>
</tr>
<tr>
<td>DATE</td>
<td>DATE</td>
</tr>
<tr>
<td>DATETIME</td>
<td>DATE</td>
</tr>
<tr>
<td>DECIMAL</td>
<td>DECIMAL</td>
</tr>
<tr>
<td>DOUBLE</td>
<td>DECIMAL(38,12)</td>
</tr>
<tr>
<td>FLOAT</td>
<td>DECIMAL(38,12)</td>
</tr>
<tr>
<td>INT</td>
<td>INTEGER</td>
</tr>
<tr>
<td>MEDIUMINT</td>
<td>INTEGER</td>
</tr>
<tr>
<td>MEDIUMTEXT</td>
<td>VARCHAR(16777215)</td>
</tr>
<tr>
<td>NCHAR(36)</td>
<td>VARCHAR(36)</td>
</tr>
<tr>
<td>NUMERIC</td>
<td>DECIMAL</td>
</tr>
<tr>
<td>NVARCHAR</td>
<td>VARCHAR</td>
</tr>
<tr>
<td>REAL</td>
<td>DECIMAL(38,12)</td>
</tr>
<tr>
<td>SET(‘A’,’B’,’C’,’D’)</td>
<td>VARCHAR(7)</td>
</tr>
<tr>
<td>SMALLINT</td>
<td>INTEGER</td>
</tr>
<tr>
<td>TEXT</td>
<td>VARCHAR(65535)</td>
</tr>
<tr>
<td>TIME</td>
<td>DATE</td>
</tr>
<tr>
<td>TIMESTAMP</td>
<td>DATE</td>
</tr>
<tr>
<td>TINYINT</td>
<td>INTEGER</td>
</tr>
<tr>
<td>TINYTEXT</td>
<td>VARCHAR(255)</td>
</tr>
<tr>
<td>VARCHAR</td>
<td>VARCHAR</td>
</tr>
<tr>
<td>YEAR</td>
<td>INTEGER</td>
</tr>
</tbody>
</table>

Non Supported Data Types
Source MySQL tables with columns of the following MySQL data types are not supported and will be ignored:
Defining the Connection Parameters
You can add a MySQL database to Attunity Compose to use as a source.

To add a MySQL source database to Attunity Compose:

1. Open the New Data Source dialog box and specify a name for it, as described in Adding Data Sources in Compose.
2. Enter the information as described in the table below.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Landing Area Settings</strong></td>
<td></td>
</tr>
<tr>
<td>Content Type</td>
<td>Choose whether the content in the landing area is Full Load, Change Processing or Full Load and Change Processing (according to the</td>
</tr>
</tbody>
</table>
### Field Description

Attunity Replicate task definition).
See also After applying changes below.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Designated By</td>
<td>Select whether the landing area is a <strong>Database</strong> or a <strong>Schema</strong>. This will depend on how the target endpoint was defined in the Attunity Replicate task. For more information, see Defining an Attunity Replicate Task.</td>
</tr>
<tr>
<td>Database Name</td>
<td>If the landing area is designated by a <strong>Database</strong>, specify the database name. This must be the same as the MySQL target database defined in the Attunity Replicate task. For more information, see Defining an Attunity Replicate Task.</td>
</tr>
<tr>
<td>Schema Name</td>
<td>Specify the schema name. This must be the same as the schema name specified in the Attunity Replicate task. For more information, see Defining an Attunity Replicate Task.</td>
</tr>
<tr>
<td>Error Mart Schema Name</td>
<td>Specify the schema where you want the data mart exception tables to be created. Data that is rejected by data quality rules will be copied to tables in the specified schema. For more information on error marts, see Defining and Managing Data Quality Rules.</td>
</tr>
<tr>
<td>After applying changes</td>
<td>If you selected <strong>Change Processing</strong> or <strong>Full Load and Change Processing</strong> as the <strong>Content Type</strong>, you can determine whether the Change Tables will be deleted or archived after the changes have been applied (to the data warehouse tables). If you select <strong>Archive the Change Tables</strong>, you also need to specify a <strong>Database name</strong> and <strong>Schema name</strong> in the designated fields.</td>
</tr>
</tbody>
</table>

#### Source Database Connection - Select this option if you want to generate the model by discovering the source database.

<table>
<thead>
<tr>
<th>Type</th>
<th>Select <strong>MySQL</strong>.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server Name</td>
<td>Specify the name or IP address of the MySQL server machine.</td>
</tr>
<tr>
<td>Port</td>
<td>Optionally, change the default port.</td>
</tr>
<tr>
<td>User Name</td>
<td>Specify your username for accessing the MySQL database. The specified user must have read/write privileges on the MySQL database.</td>
</tr>
<tr>
<td>Password</td>
<td>Specify your password for accessing the MySQL database.</td>
</tr>
<tr>
<td>Database Name</td>
<td>Specify the name of the MySQL database.</td>
</tr>
<tr>
<td>Schema</td>
<td>Specify the schema containing the source tables.</td>
</tr>
</tbody>
</table>
3. Click **Test Connection** to verify that Compose is able to establish a connection with the specified database and/or landing area.
4. Click **OK** to save your settings.
   The database is added to the list on the left side of the **Manage Databases** dialog box.

Using IBM DB2 as a Source

This section describes how to set up IBM DB2 as a source database in a Compose project.

It contains the following topics:

- **Prerequisites**
- **Data Types**
- **Defining the Connection Parameters**

**Prerequisites**

Before you begin to work with an IBM DB2 database as a source in Attunity Compose, make sure the following prerequisites have been met:

- The IBM DB2 database should be configured with the required **Data Source Permissions** and accessible from the Compose machine.
- The IBM Data Server Driver for ODBC and CLI version 10.5 must be installed on the Attunity Compose machine.

**Data Types**

The IBM DB2 database for Attunity Compose supports most IBM DB2 data types. The following table shows the IBM DB2 source data types that are supported when using Attunity Compose and the default mapping to Attunity Compose data types.

For information on how to view the data type that is mapped in the data warehouse, see the section for the data warehouse database you are using.

**Table 6.5 | Supported IBM DB2 Data Types with Mapping to Attunity Compose Data Types**

<table>
<thead>
<tr>
<th>IBM DB2 Data Types</th>
<th>Attunity Compose Data Types</th>
</tr>
</thead>
<tbody>
<tr>
<td>DATE</td>
<td>DATE</td>
</tr>
<tr>
<td>TYPE_TIMESTAMP</td>
<td>DATE</td>
</tr>
<tr>
<td>TIMESTAMP</td>
<td>DATE</td>
</tr>
<tr>
<td>TYPE_TIME</td>
<td>DATE</td>
</tr>
</tbody>
</table>
Table 6.5 | Supported IBM DB2 Data Types with Mapping to Attunity Compose Data Types

<table>
<thead>
<tr>
<th>IBM DB2 Data Types</th>
<th>Attunity Compose Data Types</th>
</tr>
</thead>
<tbody>
<tr>
<td>TYPE_DATE</td>
<td>DATE</td>
</tr>
<tr>
<td>REAL</td>
<td>DECIMAL (18,6)</td>
</tr>
<tr>
<td>DOUBLE</td>
<td>DECIMAL (18,6)</td>
</tr>
<tr>
<td>DECIMAL</td>
<td>DECIMAL</td>
</tr>
<tr>
<td>SMALLINT</td>
<td>INTEGER</td>
</tr>
<tr>
<td>INTEGER</td>
<td>INTEGER</td>
</tr>
<tr>
<td>BIGINT</td>
<td>BIGINT</td>
</tr>
<tr>
<td>VARCHAR</td>
<td>VARCHAR</td>
</tr>
<tr>
<td>WVARCHAR</td>
<td>VARCHAR</td>
</tr>
<tr>
<td>CHAR</td>
<td>VARCHAR (4000)</td>
</tr>
<tr>
<td>WCHAR</td>
<td>VARCHAR (4000)</td>
</tr>
<tr>
<td>BINARY</td>
<td>BYTE</td>
</tr>
</tbody>
</table>

Defining the Connection Parameters
You can add an IBM DB2 database to Attunity Compose to use as a source.

To add an IBM DB2 source database to Attunity Compose:
1. Open the New Data Source dialog box and specify a name for it, as described in Adding Data Sources in Compose.
2. Enter the information as described in the table below.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Landing Area Settings</strong></td>
<td></td>
</tr>
<tr>
<td>Content Type</td>
<td>Choose whether the content in the landing area is <strong>Full Load, Change Processing</strong> or <strong>Full Load and Change Processing</strong> (according to the Attunity Replicate task definition). See also <strong>After applying changes</strong> below.</td>
</tr>
<tr>
<td>Designated By</td>
<td>Select whether the landing area is a <strong>Database</strong> or a <strong>Schema</strong>. This will depend on how the target endpoint was defined in the Attunity Replicate task. For more information, see <strong>Defining an Attunity Replicate Task</strong>.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Database Name</td>
<td>If the landing area is designated by a <strong>Database</strong>, specify the database name. This must be the same as the IBM DB2 target database defined in the Attunity Replicate task. For more information, see <strong>Defining an Attunity Replicate Task</strong>.</td>
</tr>
<tr>
<td>Schema Name</td>
<td>Specify the schema name. This must be the same as the schema name specified in the Attunity Replicate task. For more information, see <strong>Defining an Attunity Replicate Task</strong>.</td>
</tr>
<tr>
<td>Error Mart Schema Name</td>
<td>Specify the schema where you want the data mart exception tables to be created. Data that is rejected by data quality rules will be copied to tables in the specified schema. For more information on error marts, see <strong>Defining and Managing Data Quality Rules</strong>.</td>
</tr>
<tr>
<td>After applying changes</td>
<td>If you selected <strong>Change Processing</strong> or <strong>Full Load and Change Processing</strong> as the <strong>Content Type</strong>, you can determine whether the Change Tables will be deleted or archived after the changes have been applied (to the data warehouse tables). If you select <strong>Archive the Change Tables</strong>, you also need to specify a <strong>Database name</strong> and <strong>Schema name</strong> in the designated fields.</td>
</tr>
</tbody>
</table>

**Source Database Connection - Select this option if you want to generate the model by discovering the source database.**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Select <strong>IBM DB2</strong>.</td>
</tr>
<tr>
<td>Server Name</td>
<td>Specify the name or IP address of the IBM DB2 server machine.</td>
</tr>
<tr>
<td>Port</td>
<td>Optionally, change the default port.</td>
</tr>
<tr>
<td>User Name</td>
<td>Specify your username for accessing the IBM DB2 database. The specified user must have read/write privileges on the IBM DB2 database.</td>
</tr>
<tr>
<td>Password</td>
<td>Specify your password for accessing the IBM DB2 database.</td>
</tr>
<tr>
<td>Database Name</td>
<td>Specify the name of the IBM DB2 database.</td>
</tr>
<tr>
<td>Schema</td>
<td>Specify the schema containing the source tables.</td>
</tr>
</tbody>
</table>

3. Click **Test Connection** to verify that Compose is able to establish a connection with the specified database and/or landing area.

4. Click **OK** to save your settings.
   The database is added to the list on the left side of the **Manage Databases** dialog box.
Using Teradata as a Source

This section describes how to set up Teradata as a source database in a Compose project.

It contains the following topics:

- Prerequisites
- Data Types
- Defining the Connection Parameters

Prerequisites

Before you can use Teradata as a source in an Attunity Compose project, make sure that the following prerequisites have been met:

- The Teradata database should be configured with the required Data Source Permissions and accessible from the Compose machine.
- Install Teradata Database ODBC Driver on the Compose machine
- Install .NET provider for Teradata version 15.11 on the Compose machine
- Install Teradata JDBC Driver on the Compose machine and copy the \tdgssconfig.jar\ and \terajdbc4.jar\ jar files to: \<Compose_Installation_Folder>\java\external

Data Types

The Teradata database for Attunity Compose supports most Teradata data types. The following table shows the Teradata source data types that are supported when using Attunity Compose and the default mapping to Attunity Compose data types.

For information on how to view the data type that is mapped in the data warehouse, see the section for the data warehouse database you are using.

Table 6.6 | Supported Teradata Data Types with Mapping to Attunity Compose Data Types

<table>
<thead>
<tr>
<th>Teradata Data Types</th>
<th>Attunity Compose Data Types</th>
</tr>
</thead>
<tbody>
<tr>
<td>DATETIME</td>
<td>DATE</td>
</tr>
<tr>
<td>DOUBLE</td>
<td>DECIMAL (p38, s12)</td>
</tr>
<tr>
<td>DECIMAL</td>
<td>DECIMAL</td>
</tr>
<tr>
<td>INT16</td>
<td>INTEGER</td>
</tr>
<tr>
<td>INT32</td>
<td>INTEGER</td>
</tr>
<tr>
<td>INTEGER</td>
<td>INTEGER</td>
</tr>
<tr>
<td>INT64</td>
<td>BIGINT</td>
</tr>
</tbody>
</table>
Table 6.6 | Supported Teradata Data Types with Mapping to Attunity Compose Data Types

<table>
<thead>
<tr>
<th>Teradata Data Types</th>
<th>Attunity Compose Data Types</th>
</tr>
</thead>
<tbody>
<tr>
<td>BYTE</td>
<td>BYTE</td>
</tr>
<tr>
<td>CHAR</td>
<td>VARCHAR</td>
</tr>
<tr>
<td>STRING</td>
<td>VARCHAR</td>
</tr>
</tbody>
</table>

Defining the Connection Parameters
You can add a Teradata database to Attunity Compose to use as a source.

To add an Teradata source database to Attunity Compose:
1. Open the New Data Source dialog box and specify a name for it, as described in Adding Data Sources in Compose.
2. Enter the information as described in the table below.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Landing Area Settings</strong></td>
<td></td>
</tr>
<tr>
<td>Content Type</td>
<td>Choose whether the content in the landing area is Full Load, Change Processing or Full Load and Change Processing (according to the Attunity Replicate task definition). See also After applying changes below.</td>
</tr>
<tr>
<td>Designated By</td>
<td>Select whether the landing area is a Database or a Schema. This will depend on how the target endpoint was defined in the Attunity Replicate task. For more information, see Defining an Attunity Replicate Task.</td>
</tr>
<tr>
<td>Database Name</td>
<td>If the landing area is designated by a Database, specify the database name. This must be the same as the Teradata target database defined in the Attunity Replicate task. For more information, see Defining an Attunity Replicate Task.</td>
</tr>
<tr>
<td>Error Mart Database Name</td>
<td>Specify the database where you want the data mart exception tables to be created. Data that is rejected by data quality rules will be copied to tables in the specified database.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>For more information on error marts, see <a href="#">Defining and Managing Data Quality Rules</a>.</td>
<td></td>
</tr>
<tr>
<td>After applying changes</td>
<td>If you selected <strong>Change Processing</strong> or <strong>Full Load and Change Processing</strong> as the <strong>Content Type</strong>, you can determine whether the Change Tables will be deleted or archived after the changes have been applied (to the data warehouse tables). If you select <strong>Archive the Change Tables</strong>, you also need to specify a <strong>Database name</strong> and <strong>Schema name</strong> in the designated fields.</td>
</tr>
</tbody>
</table>

### Source Database Connection - Select this option if you want to generate the model by discovering the source database.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Select <strong>Teradata</strong>.</td>
</tr>
<tr>
<td>Server Name</td>
<td>Specify the name or IP address of the Teradata server machine.</td>
</tr>
<tr>
<td>Port</td>
<td>Optionally, change the default port.</td>
</tr>
<tr>
<td>User Name</td>
<td>Specify your username for accessing the Teradata database. The specified user must have read/write privileges on the Teradata database.</td>
</tr>
<tr>
<td>Password</td>
<td>Specify your password for accessing the Teradata database.</td>
</tr>
<tr>
<td>Database Name</td>
<td>Specify the name of the Teradata database.</td>
</tr>
<tr>
<td>Authentication Method</td>
<td>The expected authentication method is set in an XML file by the TeraGSS program, tdgssconfigure. The following authentication methods are available:</td>
</tr>
<tr>
<td></td>
<td>» <strong>Empty</strong> - The same as omitting the keyword.</td>
</tr>
<tr>
<td></td>
<td>» <strong>TD1</strong> - Selects Teradata 1 as the authentication mechanism. User name and password are required.</td>
</tr>
<tr>
<td></td>
<td>» <strong>TD2</strong> - Selects Teradata 2 as the authentication mechanism. User name and password are required.</td>
</tr>
<tr>
<td></td>
<td>» <strong>TDNEGO</strong> - Selects one of the Authentication Mechanisms automatically based on the policy, without user involvement.</td>
</tr>
<tr>
<td></td>
<td>» <strong>LDAP</strong> - Selects Lightweight Directory Access Pro-</td>
</tr>
</tbody>
</table>
Field | Description
--- | ---
| | tocol (LDAP) as the Authentication Mechanism. The application provides the user name and password.
| | **KRB5** - Selects Kerberos (KRB5) on Windows clients working with Windows servers. The application provides the user name and password.
| | **NTLM** - Selects NT LAN Manager (NTLM) on Windows clients working with Windows servers. The application provides the user name and password.

3. Click **Test Connection** to verify that Compose is able to establish a connection with the specified database and/or landing area.
4. Click **OK** to save your settings.
   The database is added to the list on the left side of the **Manage Databases** dialog box.

### Using Amazon Redshift as a Source

This section describes how to set up Amazon Redshift as a data source in a Compose project.

It contains the following topics:

- **Prerequisites**
- **Data Types**
- **Defining the Connection Parameters**

#### Prerequisites

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 an Attunity Compose 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 Attunity Compose to work with the Amazon Redshift database.
- **Download and install the SQL client tools necessary to connect to the Amazon Redshift cluster.** Attunity Compose requires that you download a 64-bit ODBC driver.
For a list of drivers supported by Amazon Redshift, see [http://docs.aws.amazon.com/redshift/latest/mgmt/configure-odbc-connection.html](http://docs.aws.amazon.com/redshift/latest/mgmt/configure-odbc-connection.html). By default, Attunity Compose uses the Amazon Redshift (x64) driver. If you use a different driver, you must change this in the Amazon Redshift database settings in the Attunity Compose Console.

- The Amazon Redshift should be configured with the required Data Source Permissions and accessible from the Compose machine.
- Open firewall port 5439 (Amazon Redshift Cluster) for outbound communication.
- For best performance, it is strongly recommended to install Attunity Compose on an Amazon EC2 machine located in the same region as your Amazon Redshift instance.

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](http://docs.aws.amazon.com).

**Setting up SSL**

The Attunity Compose for Amazon Redshift AMI comes preinstalled with the Amazon Redshift public certificate, which it uses to establish a secure connection (SSL) to the Amazon Redshift Cluster. Since Compose always connects to Amazon Redshift using SSL, you need to replace the public certificate before it expires.

**To check the certificate’s expiration date**

- Run the following command:
  ```
  C:\Program Files\Java\jre<version>\bin>keytool -printcert -file "C:\Program Files\Attunity\Compose\java\bin\redshift-ssl-ca-cert.pem"
  ```
  The expiration date will be displayed in the "until" field:

  ![Certificate Details](image.png)

**To replace the certificate**

1. Download the Amazon Redshift public key from here:
2. Make a copy of the file and rename it to `root.crt`. Then, copy the `root.crt` file to the following directory:
   ```
   C:\Windows\System32\config\systemprofile\AppData\Roaming\postgresql\%
   ```
3. Copy the original file (`redshift-ssl-ca-cert.pem`) to the following directory:
   ```
   <product_dir>\java\bin\%
   ```
4. Open a command prompt as administrator and run the following command to register the certificate:
   ```
   <product_dir>\bin\acjs.bat server importcert redshift-ssl-ca-cert.pem
   ```
The following message should be displayed:

Status: 0 - Certificate imported

Data Types
The following table shows the Amazon Redshift data warehouse data types that are supported when using Attunity Compose and the default mapping to Attunity Compose data types.

For information on how to view the data type that is mapped in the data warehouse, see the section for the data warehouse database you are using.

<table>
<thead>
<tr>
<th>Amazon Redshift Data Types</th>
<th>Attunity Compose Data Types</th>
</tr>
</thead>
<tbody>
<tr>
<td>SMALLINT</td>
<td>INTEGER</td>
</tr>
<tr>
<td>INTEGER</td>
<td>INTEGER</td>
</tr>
<tr>
<td>BIGINT</td>
<td>BIGINT</td>
</tr>
<tr>
<td>DECIMAL</td>
<td>DECIMAL</td>
</tr>
<tr>
<td>REAL</td>
<td>DECIMAL (18,6)</td>
</tr>
<tr>
<td>DOUBLE PRECISION</td>
<td>DECIMAL (38,12)</td>
</tr>
<tr>
<td>BOOLEAN</td>
<td>INTEGER</td>
</tr>
<tr>
<td>CHAR</td>
<td>VARCHAR</td>
</tr>
<tr>
<td>VARCHAR</td>
<td>VARCHAR</td>
</tr>
<tr>
<td>DATE</td>
<td>DATE</td>
</tr>
<tr>
<td>TIMESTAMP</td>
<td>DATETIME</td>
</tr>
</tbody>
</table>

Defining the Connection Parameters
This section describes how to use an Amazon Redshift database as a data source in an Attunity Compose project.

To add an Amazon Redshift source database to Attunity Compose:
1. Open the **New Data Source** dialog box and specify a name for it, as described in **Adding Data Sources in Compose**.
2. Enter the information as described in the table below.
**Field** | **Description**
--- | ---
**Landing Area Settings** |  
Content Type | Choose whether the content in the landing area is **Full Load, Change Processing** or **Full Load and Change Processing** (according to the Attunity Replicate task definition). See also **After applying changes** below.  
Designated By | Select whether the landing area is a **Database** or a **Schema**. This will depend on how the target endpoint was defined in the Attunity Replicate task. For more information, see **Defining an Attunity Replicate Task**.  
Database Name | If the landing area is designated by a **Database**, specify the database name. This must be the same as the Amazon Redshift target database defined in the Attunity Replicate task. For more information, see **Defining an Attunity Replicate Task**.  
Schema Name | Specify the schema name. This must be the same as the schema name specified in the Attunity Replicate task. For more information, see **Defining an Attunity Replicate Task**.  
Error Mart Schema Name | Specify the schema where you want the data mart exception tables to be created. Data that is rejected by data quality rules will be copied to tables in the specified schema. For more information on error marts, see **Defining and Managing Data Quality Rules**.  
**After applying changes** | If you selected **Change Processing** or **Full Load and Change Processing** as the **Content Type**, you can determine whether the Change Tables will be deleted or archived after the changes have been applied (to the data warehouse tables). If you select **Archive the Change Tables**, you also need to specify a **Database name** and **Schema name** in the designated fields.  
**Source Database Connection** - Select this option if you want to generate the model by discovering the source database.  
Type | Select **Amazon Redshift**.  
Server Name | Specify the name or IP address of the Amazon Redshift cluster.  
Port | Optionally, change the default port.  
User Name | Specify your user name for accessing the Amazon Redshift database. The specified user must have read/write privileges on the Amazon Redshift database.
### Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Password</td>
<td>Specify your password for accessing the Amazon Redshift database.</td>
</tr>
<tr>
<td>Database Name</td>
<td>Specify the name of the Amazon Redshift database.</td>
</tr>
<tr>
<td>Schema</td>
<td>Specify the schema containing the source tables.</td>
</tr>
<tr>
<td><strong>More Options</strong></td>
<td>Click to see or hide advanced options.</td>
</tr>
<tr>
<td>Character column size in bytes</td>
<td>This should be calculated according to the largest value you are likely to store in a <code>VARCHAR</code> column. Tables in the Landing Area will be divided by the specified value (and rounded up). So, for example, if the value of <strong>Character column size in bytes</strong> is 3 (the default), both <code>VARCHAR</code> (12 bytes) and <code>VARCHAR</code> (10 bytes) will be discovered as <code>VARCHAR</code> (4 characters). <strong>See also:</strong> <a href="http://docs.aws.amazon.com/redshift/latest/dg/c_best-practices-smallest-column-size.html">http://docs.aws.amazon.com/redshift/latest/dg/c_best-practices-smallest-column-size.html</a></td>
</tr>
</tbody>
</table>

**Note** If this value is changed, existing tables will not be affected (i.e. the change will only take effect the next time the data source is discovered).

3. Click **Test Connection** to verify that Compose is able to establish a connection with the specified data warehouse.

4. Click **OK** to save your settings.
   
   The database is added to the list on the left side of the **Manage Endpoint Connections** dialog box.

### Managing Databases

You can edit and delete databases as required. The table below describes the available options.

<table>
<thead>
<tr>
<th>To</th>
<th>Do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit a database</td>
<td>In the left side of the <strong>Manage Databases</strong> dialog box, select the database that you want to edit and then click the <strong>Edit</strong> toolbar button.</td>
</tr>
<tr>
<td>Delete a database</td>
<td>In the left side of the <strong>Manage Databases</strong> dialog box, select the database that you want to delete and then click the <strong>Delete</strong> toolbar button.</td>
</tr>
</tbody>
</table>
7 | Creating and Managing the Model

This section describes how to create, import and manage the model.

The model serves as the basis for data warehouse generation in Compose. There are three way of creating the model: Use Compose to derive a tentative model by reverse engineering the source database(s) (a process also known as "discovering"); Import a model created in ERwin or create the model manually in Compose.

In this chapter:
Generating the Model
Limitations
Validating the Model
Displaying the Model
Managing the Model
Creating Expressions
Opening the Expression Builder
Defining Reusable Transformations

Generating the Model
This section explains how to generate a Business Model from a source database. You can generate the model using any of the following methods:
» Use Compose to discover the source database or landing area
» Import an ERwin model into Compose
» Create the model manually in Compose
For information about importing a model created in ERwin, see Importing the Model from ERwin.

Discovering the Source Database or Landing Area
Discovery can either be performed on the source database defined for the Attunity Replicate task or in the landing area. The decision where to perform the discovery is determined by several factors, as explained in the following table:
The source tables selected in the Attunity Replicate task contain foreign keys that you want to maintain in the Compose project.

**Note** Attunity Replicate does not support foreign key replication.

The source database defined for the Attunity Replicate task is not natively supported by Attunity Compose.

The selected source tables contain keys that are not relevant to the data warehouse (e.g. surrogate keys and business keys)

A transformation defined for the Attunity Replicate task means that not all of the columns will be replicated to the landing area. In this case, you should discover the landing area since this is the data that you eventually want to appear in your data warehouse.

**To generate the model by discovery:**

1. Open your project.
2. To generate the model from within Compose:
   a. In the **Model** panel, select **Discover** from the drop-down menu in the top right corner.
      -OR-
      In the **Manage Model** window, click the **Discover** toolbar button.
      The **Discover** dialog box opens.
   b. Select whether to discover the source database or the landing area and then click **OK**.
      Note that the suffix "/landing" denotes the landing area whereas the actual source database appears without the suffix.
3. Choose one of the following **Search for** options:
   a. To list tables only, select **Tables**.
   b. To list views only, select **Views**.
   c. To list tables and views, select **All**.

4. If you also want the internal Attunity tables to be included in the search results, select the **Show Internal Attunity Tables** check box. This may be useful for debugging, but is not usually not necessary.

5. To display all tables/views, click **Search**.

6. To only display tables/views whose names contain a specific string, type the string in the **Name** field and then click **Search**.

   The tables/views will be displayed in the **Results** list.

7. In the **Results** list, select the source tables and/or views on which to base the model or click the >> button (**Add All**) to add all of the tables in the schema.

   **Note** You can select multiple tables/views by holding down the [Shift] (sequential selection) or {Ctrl} (non-sequential selection) button.

8. To add the selected tables/views, click the > (**Add**) button.

   **Note** If you add a table that already exists in the model with the same name, then the new table is added with the name: **source_table_name_01** (or **source_table_name_02** if the name **source_table_name_01** already exists, and so on).

   If the table contains attribute domains that differ from existing ones but have the same name, they will also be appended with the _01 suffix.
9. Click OK to generate the model from the selected tables/views.

   The Generating Model from [model name] window opens.

   A progress bar indicates the current model generation progress. For each stage of the model generation process, a corresponding message appears in the Messages list.

10. After the model has been generated, click Close.

11. Repeat Steps 2-9 to discover additional sources.

Importing Entities and Mappings from Another Project

You can import entities and mappings from another project with the same data warehouse type. This is especially useful within a development environment if you need to integrate a private developer’s project with the main project.

To import entities and mappings

1. Open the Manage Model window as described in Managing the Model.
2. In the Entities toolbar, click the Import from Project button.
3. The Import from Project wizard opens.
4. In the Entities tab:
   a. Select a project from the Import from Project drop-down list.
   b. Optionally, search for specific entities.
   c. Select which entities to import.
5. Click Next to select which mappings to import.

   **Note** To create new entities and mappings if the selected entities and mappings already exist, clear the Replace existing entities and mappings check box.

   The new entities/mappings will be named <existing_name>_IMPORTED (or <existing_name>_IMPORTED_<n> if the entity/mapping is imported more than once).

6. In the Mappings tab:

   Either click Finish to import all mappings for the selected entities (the default).
   -OR-
   Select which mappings you want to import and then click Finish to import the selected entities and mappings.

   **Note** If you do not wish to import any mappings, clear the Mappings check box before clicking Finish.
Importing the Model from ERwin

In order to import a model created in ERwin, you first need to export the model from ERwin to an XML file and then copy the XML file to the Compose Server machine. Note that when you import a model from ERwin, you need to create the Mappings ETL scripts manually. You can either do this by creating global mapping as described in Managing Global Mappings below or you can create the mapping ETL directly in the Data Warehouse panel.

For more information on creating the ETL mapping(s) in the Data Warehouse panel, see Creating and Managing the Data Warehouse.

To import the model from ERwin:
1. Open your project.
2. To import a model created in ERwin, in the Model panel, select Import from ERwin from the drop-down menu in the top right corner.
   -OR-
   In the Manage Model window, select Import from ERwin from the Entities drop-down menu.
   The Import from ERwin dialog box opens.
3. Specify the full path to the ERwin XML file.
4. If you have set up global mappings, select the Use Global Mappings check box. For details, see Managing Global Mappings.
5. Select a source database and then click OK.
   The Select Tables/Views dialog box opens.
6. Continue from Step 4 in Generating the Model.

Managing Global Mappings

Before you import a data model from ERwin, you can set up the global mappings from the logical ERWin model (the entities and attributes) to the physical source database (the tables and columns). This is useful if numerous entities in your model contain the same attribute. For example, let’s assume that twenty source entities contain an attribute called "BusinessKey". In the physical source tables however, this column (which also appears in twenty tables) is called "Key". Using the Global Mappings feature, you only need to define the "Key-to-BusinessKey" mapping once instead of twenty different times.

When you import from ERwin, you can then select the Use Global Mappings check box to apply these mappings. See also Importing the Model from ERwin.

You can add, edit, and remove entity and attribute mappings. If needed, you can also change the source database referenced for the tables (if you have several different sources defined).

To manage global mappings:
1. In the Model panel, from the drop-down menu in the top right, select Global Mappings.
The **Global Mappings** window opens in the **Tables to Entities** tab.

2. Import the ERwin entities:
   a. Click **Import Entities to Mappings** toolbar button.
      The **Import Entities** dialog box opens.
   b. In the **File Path** field, enter the full path to the ERwin .xml file (on the Compose Server machine) that includes the entities you want to import.
   c. Click **OK**.

3. Verify that Attunity Compose is using the desired source database. The database name is displayed in green at the bottom right of the toolbar.
   To select a different source database:
   a. Click **Change Source Database**.
   b. In the **Set Source Database** dialog box, select a different database and then click **OK**.

4. Add new entities, edit existing entities, or remove entities as described in the following table.

<table>
<thead>
<tr>
<th>To</th>
<th>Do This</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add a new entity</td>
<td>a. In the <strong>Tables to Entities</strong> tab, click the <strong>New</strong> toolbar button.</td>
</tr>
<tr>
<td></td>
<td>b. Next to the <strong>Entity Name</strong> field, click the browse button.</td>
</tr>
<tr>
<td></td>
<td>The <strong>Unmapped Entities</strong> dialog box opens, listing only entities that have not yet been mapped.</td>
</tr>
<tr>
<td></td>
<td>c. Select an entity and click <strong>OK</strong>.</td>
</tr>
<tr>
<td></td>
<td>d. Next to the <strong>Table Name</strong> field, click the magnifying glass icon.</td>
</tr>
<tr>
<td></td>
<td>The <strong>Find Table for [Entity Name]</strong> dialog box opens for the selected entity.</td>
</tr>
<tr>
<td></td>
<td>e. From the <strong>Tables</strong> drop-down list on the left, select the table to map to.</td>
</tr>
<tr>
<td></td>
<td>f. Click <strong>OK</strong>.</td>
</tr>
<tr>
<td></td>
<td>Attunity Compose populates the <strong>Table Schema</strong> field automatically, based on the table you selected.</td>
</tr>
<tr>
<td></td>
<td>g. Repeat these steps for all unmapped entities.</td>
</tr>
</tbody>
</table>

| Edit an entity | a. Move the mouse cursor over the entity and click the **Edit** button (pencil icon) that appears on the right. |
|               | b. Make the required changes and click **OK**.                         |

| Delete an entity | a. Select the entity.         |
|                 | b. In the **Entities** toolbar, click **Delete**.                      |
|                 | c. When prompted to confirm the deletion, click **Yes**.               |
To | Do This
---|---
Search for an entity | In the **Search** look-up field, start typing. Attunity Compose only displays entities that match the search string.

5. **Add**, **edit**, or **remove** attributes as described in the table below.

<table>
<thead>
<tr>
<th>To</th>
<th>Do This</th>
</tr>
</thead>
</table>
| **Add a new attribute** | a. In the **Columns to Attributes** tab, click the **New** toolbar button.  
b. Provide a name and description (optional) for the attribute and the column.  
c. Click **OK**. |
| **Search for an attribute** | In the **Search** look-up field, start typing. Attunity Compose only displays attributes that match the search string. |
| **Edit an attribute** | a. Move the mouse cursor over the attribute and click the **Edit** button (pencil icon) that appears on the right.  
b. Make the required changes and click **OK**. |
| **Remove an attribute** | a. Select the attribute.  
b. Click the **Delete** toolbar button.  
c. When prompted to confirm the deletion, click **Yes**. |

6. Click **Close**.

**Limitations**

- When Amazon Redshift is the data warehouse type, attribute names that contain the open parenthesis character "(" are not supported. If any of your attribute names contain the "(" character, you should remove it before creating the data warehouse tables.
  
  For information on renaming attribute names, see [Add an attribute to all Satellite tables and the Hub table](#).

- Discovering new tables does not affect existing entities in the model, even if there is a relationship between the new entity and one of the existing entities. For example, in the source database, Table 1 has a Foreign Key that points to Table 2. If Table 1 is added to the model and then Table 2 is added later, Table 1 will not be updated to contain the required Foreign Key.

- The data warehouse needs to be "adjusted" when deleting a relationship/attribute from
the model and then adding the same relationship/attribute back to the model. However, the "Adjust" operation deletes the data from the corresponding data warehouse column.

Validating the Model

Once you have generated the model, you can easily check that it is valid. For example, for a model to be valid, each of the tables must have a Business Key.

**Note** Validating the model does not recalculate expressions for historical data that has changed.

**To validate the model:**
1. Either click the **Validate** button in the bottom right of the **Model** panel.
   - OR -
   Select **Validate** from the drop-down menu in the top right of the **Model** panel.
   The **Validate Model** window opens.
   If the model is valid, a message will confirm the model’s validity. If the model is not valid, a list of invalid tables/views will be displayed.
   A message indicating why the entity is invalid will be displayed in the **Message** column.
   To resolve the issue, click the **Edit Entities** button to the right of the entity.
   The **Edit Model** window opens showing the invalid entity.
2. Resolve the issue (in this case, by adding a Business Key) and then click **Close**.
   A message will confirm the model’s validity.
3. Click **Close** to close the **Validate Model** window.

Displaying the Model

Displaying the model is a good way to see the relationships between the various tables and/or views in your model.

**To display the model:**
1. Either click the **Display** button in the bottom right of the **Model** panel.
   - OR -
   Select **Display** from the drop-down menu in the top right of the **Model** panel.
   The **Display Model** window opens in Diagram view.
2. In Diagram view, the following display options are available:

**Note** You can select multiple entities by clicking them while holding down the [Ctrl] keyboard button.

- **Zoom** - Increase or decrease the magnification using the slider at the top right of the screen. Click the button to the right of the slider to restore the default size.
- **Search** - The ability to search for entities is particularly useful in a large model. To search for an entity, type a search string in the Search box. Compose lists the names of entities that match the search string. Select the desired entity.

- **Drag the diagram** - In addition to zooming, you can also drag the diagram by clicking the space around the diagram and dragging. This is useful for very large diagrams where zooming out would render the text unreadable. The guide at the bottom right of the dialog box shows you which part of the diagram is currently displayed.
» **Show/Hide business keys** - Select/Clear the **Keys** check box at the top of the dialog box.

» **Show/Hide attributes** - Select/Clear the **Attributes** check box at the top of the dialog box.

» **Change the Diagram Direction** - Select one of the available options from the **Direction** drop-down list at the top of the dialog box.

» **Create a relationship to another entity** - See Adding Relationships via the Display Model Dialog Box.

» **Select path** - To highlight the path to which an entity belongs, either hover your mouse cursor over the entity or right-click the entity and select **Select Path**.

» **Select neighbors** - Right-click an entity and select this option to highlight the entity’s neighbors.

» **Select all** - Right-click an entity and select this option to highlight all entities in the model.

» **Edit** - Either double-click the entity or right-click an entity and select the **Edit** option to edit the entity’s attributes.

» **Lineage** - Right-click an entity and select this option to show/hide the entity’s lineage. For more information on lineages, see Lineage and Impact Analysis.

» **Hide this node** - Right-click an entity and select this option to show/hide the entity. To show the entity, click the **Hidden Nodes** box in the left of the dialog box.

» **Hide selected nodes** - Right-click an entity and select this option to show/hide selected entities. To show the hidden entities, click the **Hidden Nodes** box in the left of the dialog box.

» **Invert selection** - Right-click an entity and select this option to highlight all entities except the selected entity.

» **Hide non-selected nodes** - Right-click an entity and select this option to show/hide non-selected entities. To show the hidden entities, click the **Hidden Nodes** box in the left of the dialog box.

3. To switch to Tree view, click the **Tree View** tab. In Tree view, the following display options are available:

» **Expand/Collapse** - Click the arrow to the left of a table to see its columns or related tables. To show or hide all sub-tables and table columns, click the **Expand All/Collapse All** buttons at the top of the **Tree View** tab.

» **Lineage** - To see an entity’s lineage, hover your mouse over a table or column and
then click the link icon that appears to its right.

4. To search for a specific entity, enter a part of the entity’s name in the Search box. Entities that match the search string will be highlighted.

Managing the Model

You can edit the model according to your needs. The following editing options are available:

- Managing Entities
- Creating and Managing Relationships
- Managing Attributes
- Bulk Editing History Types and Satellite Numbers
- Lineage and Impact Analysis

There are two ways of editing a model in Compose:

- **In the Manage Model window** - Editing the model in the Manage Model window is preferable if you need to make several changes to the model as it provides access to all of the model’s entities and attributes. To display the results of your changes, open the Model Display window as described in Displaying the Model.

- **From the Model Display** - Editing the model from the Model Display window is convenient if you only need to edit one or two entities. Another advantage of this method is that it allows you to see the result of your changes (in the entity relationship diagram) immediately.

**To open the Manage Model dialog box from the Model panel:**

1. Click the Manage button at the bottom left of the Model panel or click the Entities link in the Model panel.
   
   The Manage Model dialog box opens.

2. Edit the model according to the descriptions below.
To open the Manage Model dialog box from the Model Display window:

1. Open the Model Display window as described in Displaying the Model.
2. Double-click the entity you want to edit.
   The Manage Model dialog box opens.
3. Edit the model according to the descriptions below.

The Manage Model Window

The Manage Model window is split into two tabs: The Logical Model tab and the Physical Model tab. The Logical Model tab shows the entities and attributes as they appear in the model whereas the Physical Model tab provides a preview of the actual tables (and columns) that will be created in the data warehouse. So, for example, although the Categories table appears as a single entity in the Logical Model tab, it will appear as two tables (TDWH_Catergories_HUB and TDWH_Catergories_S01) in the Physical Model tab. The reason for this is because the logical Categories entity contains both Type 1 and Type 2 attributes. Type 1 attributes will be created as columns in the HUB table while Type 2 attributes will be created as columns in the Satellite table (S01). For more information on Type 1 and Type 2 attributes, see History.

All editing tasks are performed in the Logical Model tab, except for the following tasks which are performed in the Physical Model tab:

- Designate a Distribution Key Column (Amazon Redshift only)
- Designate Primary Index Column(s) (Teradata only)

For more information, see Designating Distribution Key and Primary Index Columns.

Managing Entities

You can add, edit and remove entities from your model as described in the table below.

| Note | All of the options available in the toolbar are also available from the drop-down menu in the toolbar. This is useful when you reduce the window size, since some of the toolbar buttons - or all of them depending on how small you make the window - will be hidden. The only button that will not be hidden regardless of the eventual window size is the drop-down menu button. |

<table>
<thead>
<tr>
<th>Table 7.1</th>
<th>Entity Editing Options</th>
</tr>
</thead>
<tbody>
<tr>
<td>To</td>
<td>Do This</td>
</tr>
<tr>
<td>Add an entity</td>
<td>1. Click the New Entity button in the Entities toolbar.</td>
</tr>
<tr>
<td></td>
<td>2. Provide a name and description (optional) for the entity and then click OK.</td>
</tr>
<tr>
<td>Edit an entity</td>
<td>1. Select the entity you want to edit and then click the Edit button</td>
</tr>
</tbody>
</table>
Table 7.1 | Entity Editing Options

<table>
<thead>
<tr>
<th>To</th>
<th>Do This</th>
</tr>
</thead>
<tbody>
<tr>
<td>To add an entity</td>
<td>1. Click the New button in the Entities toolbar.</td>
</tr>
<tr>
<td></td>
<td>2. Edit the entity’s name and description (optional) and then click OK.</td>
</tr>
<tr>
<td>Remove an entity</td>
<td>1. Click the Delete button in the Entities toolbar.</td>
</tr>
<tr>
<td></td>
<td>2. When prompted to confirm the deletion, click Yes.</td>
</tr>
<tr>
<td>Duplicate an entity</td>
<td>1. Select the entity you want to duplicate and then click the Duplicate button in the Entities toolbar.</td>
</tr>
<tr>
<td></td>
<td>2. Edit the entity’s name and description (optional) and then click OK.</td>
</tr>
<tr>
<td></td>
<td>The duplicated entity is added to the Entities list.</td>
</tr>
<tr>
<td>Import entities from another project</td>
<td>See Importing Entities and Mappings from Another Project.</td>
</tr>
<tr>
<td>Import entities from ERwin</td>
<td>See Importing the Model from ERwin.</td>
</tr>
</tbody>
</table>

Managing Attributes
You can add, edit and remove attributes as required. All attributes in the model belong to the Attributes Domain. When adding a new attribute, you can either select an existing attribute from the Attributes Domain or create a new Attributes Domain. Both of these options are described in the table below.

Table 7.2 | Attribute Editing Options

<table>
<thead>
<tr>
<th>To</th>
<th>Do This</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add an attribute from the attributes domain</td>
<td>1. Click the New Attribute button in the Attributes toolbar.</td>
</tr>
<tr>
<td></td>
<td>The New Attribute dialog box opens.</td>
</tr>
<tr>
<td></td>
<td>2. To designate the attribute as a business key, select the Key check box.</td>
</tr>
<tr>
<td></td>
<td>3. From the Attribute domain drop-down list, select the attribute domain you wish to add.</td>
</tr>
<tr>
<td></td>
<td>4. To edit the selected attribute domain on-the-fly, click the edit button located after the Attribute domain drop-down list. This will open the Edit - AttributeDomainName dialog box. Then, continue from Step 2 in Edit an attribute domain.</td>
</tr>
<tr>
<td></td>
<td>5. In the Attribute name field, optionally change the default instance name for the attribute domain.</td>
</tr>
</tbody>
</table>
You can create multiple instances of a single Attribute Domain. This is especially useful if you want to use the same Attribute Domain across multiple tables, with each "instance" having its own unique name. This also allows you to edit the properties of each attribute without affecting the other attributes, despite all the Attribute Domain instances sharing a common Attribute Domain. For example, if the Attribute Domain name is "ID", you could create one instance for it in the "Categories" entity named "CategoryID" and another instance in the "Employees" entity named "EmployeeID". If, however, you edit the parent Attribute Domain attribute, all instances of that attribute will be updated as well.

6. To add a prefix to the attribute name, enter the desired prefix in the **Prefix** field.

Adding a prefix to an attribute name allows you to add multiple instances of the same attribute domain. For example, the attribute "Employee" could become two different attributes: "ReportsTo_Employee" and "HiredBy_Employee".

7. Set the **History Type** and Satellite number. When the **History Type** is set to 2, a new record will be created in the data warehouse each time an attribute value changes.

8. In the **Satellite/Hub** field, optionally change the satellite number. Note that the satellite number can only be changed when the **History Type** is set to 2. For an explanation of why this is so, see *The Manage Model Window*.

9. To add an expression, click the **fx** button located after the **Expression** field and then continue from *Creating Expressions*.

10. Click **OK** to save your settings.

| Create a new attribute domain and add it to the model | 1. Click the **New Attribute** button in the **Attributes** toolbar. The **New Attribute** dialog box opens.  
2. To designate the attribute as a business key, select the **Key** check box.  
3. Click the plus sign to the right of the **Attribute domain** drop-down list. The **New Attribute Domain** dialog box opens.  
   a. Specify a **Name** for the attributes domain.  
   b. From the **Type** drop-down list, select one of the available data types. |

---

To Do

This is especially useful if you want to use the same Attribute Domain across multiple tables, with each "instance" having its own unique name. This also allows you to edit the properties of each attribute without affecting the other attributes, despite all the Attribute Domain instances sharing a common Attribute Domain. For example, if the Attribute Domain name is "ID", you could create one instance for it in the "Categories" entity named "CategoryID" and another instance in the "Employees" entity named "EmployeeID". If, however, you edit the parent Attribute Domain attribute, all instances of that attribute will be updated as well.

6. To add a prefix to the attribute name, enter the desired prefix in the **Prefix** field.

Adding a prefix to an attribute name allows you to add multiple instances of the same attribute domain. For example, the attribute "Employee" could become two different attributes: "ReportsTo_Employee" and "HiredBy_Employee".

7. Set the **History Type** and Satellite number. When the **History Type** is set to 2, a new record will be created in the data warehouse each time an attribute value changes.

8. In the **Satellite/Hub** field, optionally change the satellite number. Note that the satellite number can only be changed when the **History Type** is set to 2. For an explanation of why this is so, see *The Manage Model Window*.

9. To add an expression, click the **fx** button located after the **Expression** field and then continue from *Creating Expressions*.

10. Click **OK** to save your settings.

| Create a new attribute domain and add it to the model | 1. Click the **New Attribute** button in the **Attributes** toolbar. The **New Attribute** dialog box opens.  
2. To designate the attribute as a business key, select the **Key** check box.  
3. Click the plus sign to the right of the **Attribute domain** drop-down list. The **New Attribute Domain** dialog box opens.  
   a. Specify a **Name** for the attributes domain.  
   b. From the **Type** drop-down list, select one of the available data types. |
Table 7.2 | Attribute Editing Options

<table>
<thead>
<tr>
<th>To</th>
<th>Do This</th>
</tr>
</thead>
<tbody>
<tr>
<td>c.</td>
<td>If the selected data type requires further configuration, additional fields will be displayed. For example, when Decimal is selected, the <strong>Length</strong> and <strong>Scale</strong> fields will be displayed. Set the values as desired.</td>
</tr>
<tr>
<td>d.</td>
<td>Optionally, specify a <strong>Description</strong>.</td>
</tr>
<tr>
<td>e.</td>
<td>Click <strong>OK</strong> to add the newly created attribute domain to the <strong>Attribute domain</strong> field and close the <strong>New Attribute Domain</strong> dialog box.</td>
</tr>
<tr>
<td>4.</td>
<td>Continue from <strong>Step 5 in Add an existing attribute domain</strong> above.</td>
</tr>
</tbody>
</table>

**Note**: You can also add new attribute domains via the **Manage Attribute Domains** window. For more information, see **Managing the Attributes Domain**.

<table>
<thead>
<tr>
<th>Add a relationship</th>
<th>You can use the <strong>Add to all Satellites and Hub</strong> option to define the same Primary Index for the Hub table and all Satellite tables.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>➤ Select the desired attribute and then click the <strong>Add to all Satellites and Hub</strong> toolbar button.</td>
</tr>
<tr>
<td></td>
<td>The attribute is added to the Hub table and to all the Satellite tables.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Edit an attribute</th>
<th><strong>Method 1:</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1. Select the attribute you want to edit and then click the <strong>Edit</strong> button in the <strong>Attributes</strong> toolbar.</td>
</tr>
<tr>
<td></td>
<td>The <strong>Edit - AttributeName</strong> dialog box opens</td>
</tr>
<tr>
<td></td>
<td>2. Continue from <strong>Step 2 of Add an attribute from the attributes domain</strong> above.</td>
</tr>
<tr>
<td></td>
<td><strong>Method 2:</strong></td>
</tr>
<tr>
<td></td>
<td>1. Double-click the attribute you want to edit.</td>
</tr>
<tr>
<td></td>
<td>The values in the attribute row become editable.</td>
</tr>
<tr>
<td></td>
<td>2. Continue from <strong>Step 2 of Add an attribute from the attributes domain</strong> above.</td>
</tr>
</tbody>
</table>

| Bulk edit history types and satellite numbers | See **Bulk Editing History Types and Satellite Numbers**.                                                                                                                                            |
Table 7.2 | Attribute Editing Options

<table>
<thead>
<tr>
<th>To</th>
<th>Do This</th>
</tr>
</thead>
<tbody>
<tr>
<td>Show an attribute’s lineage</td>
<td>See Lineage and Impact Analysis.</td>
</tr>
</tbody>
</table>
| Remove an attribute | 1. Select the attribute(s) you want to delete.  
2. Click the Delete button in the Attributes toolbar.  
3. When prompted to confirm the deletion, click Yes. |
| Change the attribute order | Select the attribute you want to move and use the Move Up/Move to Top and Move Down/Move to Bottom toolbar buttons to move the attribute. |
| Manage the Attributes Domain | See Managing the Attributes Domain. |
| Create an expression for an attribute | See Add an attribute from the attributes domain or Edit an attribute above. |
| Export the attributes to a CSV file | Select an entity from the Entities list on the left of the Manage Model window and then select Export to CSV from the drop-down menu in the Attributes toolbar. Depending on your browser settings, you will either be prompted to download the <entityname>_Attributes.csv file or it will be downloaded to your default Downloads location. |

Setting up Derived Attributes

Derived attributes are attributes whose data is "derived" from other attributes. For example, lets assume that the OrderDetails entity contains the attributes Quantity and UnitPrice but does not contain the attribute TotalPrice. To gain better insight into the annual sales figures, the organization would like to add the TotalPrice attribute to the model and derive its data from the Quantity and UnitPrice attributes.

Assuming that the Northwind sample database is the model’s source, this could easily be done as follows:

1. Add the TotalPrice attribute domain to the model as described in Managing Attributes.
2. After finalizing the model, create the data warehouse tables as described in Creating the Data Warehouse Tables.
3. Click the OrderDetails mapping as described in Editing Column Mappings.

Note that the TotalPrice attribute has no mapping as it was added after the Northwind source was discovered:
4. Open the Expression Builder by clicking the $fx$ icon to the right of the \textit{TotalPrice} column name. Then, in the Expression Builder, add the \textit{Quantity} and \textit{UnitPrice} columns to create the following expression:

$$\text{Quantity} \times \text{UnitPrice}$$

For more information on creating expressions, see \textit{Creating Expressions}.

5. Click \textbf{OK} to close the Expression Builder and save the expression.

The \textit{Quantity} and \textit{UnitPrice} landing area columns are now mapped to the \textit{TotalPrice} staging column. Notice that the mapping lines are gray, indicating that the mapping is the result of an expression.

Hovering the mouse cursor over the gray lines highlights the derived column (\textit{TotalPrice}) and the columns from which its data is derived (\textit{Quantity} and \textit{UnitPrice}).
Managing the Attributes Domain

The Attributes Domain provides a list of all the attributes available in the Compose model, as well as their data type. You can add, edit and delete attributes according to your data warehousing needs. The Attributes Domain also allows you to see which entities each attribute belongs to, as a single attribute may be present in several entities.

To manage the Attributes Domain

1. From the drop-down menu in the top right of the Model panel, select Attributes Domain.
2. Add, delete and edit attributes as describe in the table below.

<table>
<thead>
<tr>
<th>To</th>
<th>Do This</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add an attributes domain</td>
<td>1. Click the New Attributes Domain toolbar button. The New Attribute Domain dialog box opens.</td>
</tr>
<tr>
<td></td>
<td>2. In the Name field, specify a name for the attribute.</td>
</tr>
<tr>
<td></td>
<td>3. From the Type drop-down list, select one of the available data types.</td>
</tr>
<tr>
<td></td>
<td>4. If the selected data type requires further configuration, additional fields will be displayed. For example, when Decimal is selected, the Length and Scale fields will be displayed. Set the values as desired.</td>
</tr>
<tr>
<td></td>
<td>5. Optionally specify a Description.</td>
</tr>
<tr>
<td></td>
<td>6. Click OK to add the attribute and close the New Attribute Domain dialog box.</td>
</tr>
<tr>
<td>Edit an attribute domain</td>
<td>1. Select the desired attribute and then click the Edit toolbar button. The Edit: Name dialog box opens.</td>
</tr>
<tr>
<td></td>
<td>2. Edit the attribute as described in steps 2-6 of Add an attributes domain above.</td>
</tr>
<tr>
<td></td>
<td>Note that the Edit: Name dialog box also contains a Used in Entities list. Knowing which entities the attribute is used in may affect the type of changes you make, as the planned changes may not be appropriate for all</td>
</tr>
</tbody>
</table>
To Do This

Remove an attribute

| 1. | Select the attribute you want to delete and then click the **Delete** toolbar button. |
| 2. | When prompted to confirm the deletion, click **Yes**. |

Creating and Managing Relationships

Similar to a foreign key, a relationship "attribute" is a special type of attribute that points to another entity in the same model. Typically, the relationship replaces the key attributes that connect an entity to a related entity. You can add, edit and delete relationships as required.

Possible reason for creating relationships are as follows:

- If your model is derived from the landing area (as opposed to the source database(s)), the model will be created without any relationships
- Ensure data integrity between related entities

You can create relationship from the **Manage Model** dialog box or from the **Display Model** dialog box. Both of these methods are described below.

**Adding Relationships via the Manage Model Dialog Box**

1. Click the **Manage** button in the bottom left of the **Model** panel.
   The **Manage Model** dialog box opens.
2. Select an Entity in the **Entities** list.
3. Click the **Add Relationship** button in the **Attributes** toolbar.
   The **Add Relationship From: Name** dialog box opens.
4. From the **Add Relationship to Entity** drop-down list, select the entity to which you want to create a relationship.
5. If the originating entity contains attributes that were foreign keys in the source database, you can replace these attributes with Business Key attributes of the associated entity.
   **To do this:**
   a. Select the **Replace Existing Attribute(s)** check box.
      The left column shows the **Business Key Attributes of the Associated Entity**.
   b. From the **Attributes of Originating Entity** drop-down list on the right, select an attribute from the originating entity that was meant to be a foreign key.
6. If you want the relationship attribute to be a Business Key, select the **Business Key** check box. This option will only be displayed if the entity target can be designated as a Business Key.
7. Set the **History Type**.
8. Set a **Satellite Number**.

**Note**  Since the satellite number for Business Keys must be "0", the option to change the satellite number is unavailable when the **Business Key** check box is selected.

10. Optionally, enter a description.
11. Click **OK** to save your settings.

### Adding Relationships via the Display Model Dialog Box

1. Click the **Display** button in the bottom left of the **Model** panel.
   The **Display Model** dialog box opens.
2. Select one of the following methods:
   - **Method 1**: Right-click an entity and select **Add Relationship**.  
     The **Add Relationship From: Name** dialog box opens.
   - **Method 2**: Right-click an entity and select **Set as Relationship Source**. This method is useful if you need to search your model for the relationship target entity (since the source entity remains selected while you search).
   - **Method 3**: Select two entities by clicking them while holding down the [Ctrl] key. Then, right-click one of the entities and select the desired relationship from the context menu (according to the entity that you want to be the relationship source), as shown in the following example:

   ![Diagram of relationships]

3. If you selected **Method 2**, continue below. If you selected **Method 1**, continue from Step 4 in **Adding Relationships via the Manage Model Dialog Box**. If you selected **Method 3**, continue from Step 5 in **Adding Relationships via the Manage Model Dialog Box**.
4. Right-click the relationship target entity and select **Relationship Target for Relationship Source Name**.
   The **Add Relationship: Name** dialog box opens with the relationship target entity
already selected.
5. Continue from Step 5 in Adding Relationships via the Manage Model Dialog Box.

Deleting Relationships

1. Click the Manage button in the bottom left of the Model panel.

The Manage Model dialog box opens.
2. Select the relationship attribute you want to delete.
3. Click the Delete button in the Attributes toolbar.

The Delete Relationship dialog box opens.
4. To restore an attribute that was replaced when the relationship was created, select the Restore original attribute(s) check box.

For more information about replacing attributes, see Step 10 in Adding Relationships via the Manage Model Dialog Box.
5. Click Yes to delete the relationship attribute.

Bulk Editing History Types and Satellite Numbers
Use the Bulk Edit feature to edit the History type and Satellite number of multiple attributes.

To bulk edit history types and satellite numbers:
1. Select the attributes whose History type and/or Satellite number you want to change and click the Bulk Edit toolbar button.
2. In the Bulk Edit dialog box, change the History type and/or Satellite number as required.
3. Click OK close the Bulk Edit dialog box and save your settings.

Lineage and Impact Analysis
Before editing an entity or attribute, you may want to see which other entities/attributes in the entity’s/attribute’s lineage will be impacted by the change. For example, removing the "Discount" attribute from a table will affect the "Total Price". Additionally, a single attribute may have different names depending on its location.

Places where you can view lineage in Compose:
- The Manage Model window described below.
- The Display Model window described in Displaying the Model.
- When editing a data mart. For more information, see Managing Data Marts.
To view the lineage of an entity or attribute:

1. Click the Manage button in the bottom left of the Model panel.
   The Manage Model dialog box opens.
2. Display the lineage as described below

### Table 7.3 | Lineage Options

<table>
<thead>
<tr>
<th>To</th>
<th>Do This</th>
</tr>
</thead>
<tbody>
<tr>
<td>Show an entity’s lineage</td>
<td>Select the entity and select <strong>Show Lineage</strong> from the drop-down menu in the Entity toolbar.</td>
</tr>
<tr>
<td>Show an attribute’s lineage</td>
<td>Select the attribute and click the <strong>Show Lineage</strong> button in the Attribute toolbar.</td>
</tr>
</tbody>
</table>

**Adding Date and Time Entities to your Model**

Compose provides built-in Date and Time entities that you can add to your model. This facilitates access to all attributes of date and time (such as day of the week, quarter, and so on) both in the BI reports and when creating transformations in the data mart.

The Date entity contains a record for every day. Dates in the Date entity range from January 1st 1900 to December 31st 2099.

The Time entity contains all the hours and minutes in a 24 hour period. When you create the data warehouse tables, the Date and Time entities are automatically populated with relevant data. You can view this data as described in Viewing the Data Warehouse Tables.

Both the date and the time values are presented in multiple formats (e.g. 12 hour format or 24 hour format), allowing you to choose which format will be displayed in your BI reports. Other format include abbreviated forms of date and time, different month/year/day formats (e.g. 12/31/2017 as opposed to 2017-12-31), and so on.

You can either add the entities to a new project (before you create the Data Warehouse tables) or to an existing project. If you add them to an existing project’s model, you will also need to validate and adjust the Data Warehouse as described in Validating the Data Warehouse.

You can even add custom date and time attributes to the entities from the tables in your landing area. For example, if one of your source tables lists all the working days and non-working days, you can add an "Is Working Day" attribute to the Date entity and then load it from the relevant source table. Just like regular entities, Compose knows how to merge the incoming data of working and non-working days into the existing Date entity.

For an explanation of how to add attributes to an entity, see Managing Attributes.
You cannot add relationships to the Date and Time entities. However, every date and time attribute has an implicit relationship to the Date and Time dimensions, which allows you to select the relevant dimension when creating your star schema in the data mart.

For information on working with Date and Time dimensions in the data mart, see Creating and Managing Data Marts.

**Note** For all of the supported data sources except Oracle, you can add both Date and Time entities to your model. If you are using Oracle as your data source, you can only add the Date entity to your model. This is because Oracle does not have a data type specifically for Time.

---

**To add Date and Time entities to your model:**

1. Open the desired Compose project.
2. From the drop-down menu in the top right of the Model panel, select **Add Date and Time** entities or **Add Date Entity** if your data source is Oracle (see **Note** above).
3. When prompted to confirm the action, click **Yes**.
4. The Date and Time entities will be added to your model. By default, the Date and Time entities are hidden from the model display (as they are not related to other entities in your model). If you want to show them anyway, select the **Date and Time** model check box in the Data Warehouse Model window.
5. For information about displaying the model, see Displaying the Model.

**Note** You can also delete the Date and/or Time entities if you no longer require them and add them again later.

---

**Designating Distribution Key and Primary Index Columns**

In the **Physical Model** tab, you can perform the following tasks:

- Designate a Distribution Key column (Amazon Redshift data warehouse)
- Designate Primary Index columns (Teradata data warehouse)

**If your data warehouse is Amazon Redshift:**

Select one of the following from the **Distribution Style** drop-down list above the selected table: **Even, Key** or **All**.

If you selected **Key**, in the **Distribution Key** column, select which column will be the designated distribution key.

For more information on distribution styles, see:

http://docs.aws.amazon.com/redshift/latest/dg/c_choosing_dist_sort.html
If your data warehouse is Teradata:

Select one of the following from the Primary Index drop-down list above the selected table: Unique Primary Index, Non Unique Primary Index or No Primary Index.

If you selected Unique Primary Index or Non Unique Primary Index, in the Primary Index column, select which column(s) will make up the Primary Index.

For more information on designating Primary Index columns, see:
http://www.teradatawiki.net/2013/08/Teradata-Primary-Index.html

Creating Expressions

Compose allows you to create data transformations in several different places according to your needs. A transformation can either be a filter (i.e. excluding certain data) or an expression (i.e. manipulating a single record). The table below lists the places where transformations can be created and provides reasons for creating the transformation in each of the specified places.

Table 7.4 | Places where Transformations can be Created

<table>
<thead>
<tr>
<th>Where the Transformation is Created</th>
<th>Reasons to Create a Transformation There</th>
<th>When the Transformation is Applied</th>
</tr>
</thead>
<tbody>
<tr>
<td>Replicate</td>
<td>Filtering large amounts of data that is not needed for the data warehouse (in the present or the future)</td>
<td>Before the data reaches the landing area.</td>
</tr>
<tr>
<td></td>
<td>Obfuscation due to regulatory reasons or internal policies</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Data type conversion (e.g. converting a source data type that is not supported on the data warehouse platform)</td>
<td></td>
</tr>
<tr>
<td>Model</td>
<td>The default location if you are not sure where to put it</td>
<td>Between the staging area and the data warehouse.</td>
</tr>
<tr>
<td></td>
<td>General business logic</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Needed for several sources or several data marts</td>
<td></td>
</tr>
<tr>
<td>Data Warehouse</td>
<td>Specific source preparation</td>
<td>Between the landing area and the staging area.</td>
</tr>
<tr>
<td></td>
<td>Needed for merging several sources</td>
<td></td>
</tr>
<tr>
<td>Data Mart</td>
<td>Specific to a data mart</td>
<td>Between the data warehouse and the data mart.</td>
</tr>
<tr>
<td></td>
<td>Managed by a data mart data specialist</td>
<td></td>
</tr>
</tbody>
</table>
See also Defining Reusable Transformations.

The following topics describe the Expression Builder:

- Opening the Expression Builder
- Expression Builder Overview
- Building Expressions
- Testing Expressions

Opening the Expression Builder

The Expression Builder enables you to create a transformation without needing to type anything manually.

The Expression Builder can be opened in several places, depending on your needs. For more information about where to create a transformation, see the table Places where Transformations can be Created.

Expression Builder Overview

The following section provides an overview of the Expression Builder functionality.
The Expression Builder consists of the following panels:

» **Tabs on the left of the Expression Builder**: These tabs contain elements that you can add to an expression. Select elements and add them to the Build Expression pane to create an expression. For more information, see Building Expressions.

The following tabs are available:

» **Parameters** - Only displayed when opening the Expression Builder from within the Reusable Transformations > Edit Transformation dialog box.

For information on reusable transformations, see Defining Reusable Transformations below.

» **Input Columns/Input Attributes** - Columns/attributes that can be used to build your expression.

» **Transformations** - Contains a list of reusable transformations. The tab is not displayed if no reusable transformations have been defined.

For information on reusable transformations, see Defining Reusable Transformations below.

» **Operators** - Operators that can be used to build your expression.

» **Functions** - Functions that can be used to build your expression.

**Note** The Operators and Functions displayed in the Expression Builder use SQL format. As each data warehouse has its own implementation of SQL, the data warehouse defined in your Compose project determines which Operators and Functions will be available. For an explanation of the available Operators and Functions, refer to the documentation for your data warehouse.

» **Build Expression Pane**: The Build Expression pane is where you build your expression. You can add elements, such as columns or operators to the panel as well as type all or part of the expression. For more information, see Building Expressions.

» **Parse Expression Pane**: This pane displays the parameters for the expression. After you build the expression, click Parse Parameters to list the expression parameters. You can then edit the parameters, enter a value for each of the parameters and associate attributes with them. For more information, see Parsing Expressions.

» **Test Expression Pane**: This pane 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 Testing Expressions.

**Building Expressions**

The first step in using the Expression Builder is to build an expression in the Build Expression pane.

**To build an expression:**

1. Hover the mouse cursor over the element that you want to add to your expression (expressions usually start with an Input Column) and click the arrow that appears to
2. Add **Operators** additional **Input Columns** and **Functions** as required.

**Note** To add operators to your expression, you can use the Operator tab on the left or the Operator buttons located above the Build Expression pane or any combination of these.

**Example:**

To create an expression that combines the FirstName name and LastName columns, do the following:

1. Add the FirstName **Input Column** to the Build Expression pane.
2. Assuming that Microsoft SQL Server is the data warehouse, in the Operator toolbar above the Build Expression pane, click the concatenate (+) operator.
3. Then add a space between single quote characters and click the concatenate (+) operator again.
4. Add the LastName **Input Column** to the Build Expression pane.

   The expression would look like this:

   ```
   ${FirstName}+' '+${LastName}
   ```

**Parsing Expressions**

When you add operators to the expression, the expression’s parameters are usually added automatically to the Parse Expression pane. However, when you complete your expression or edit it, you may need to parse the expression see all of the parameters.

**To parse the expression parameters:**

- Click the Parse Expression button below the Build Expression pane.

  If the expression is *not* valid, a red error message will appear at the bottom of the Expression Builder window.

  If the expression is valid, the expression parameters and attributes (Input Columns) will be displayed in the in the Parse Expression pane. See the figure **Test Expression**.

**Editing Parameter Names**

By default, the parameter name is the same as the input column name. However, you can change the parameter name as needed and then associate it with an input column. This is useful, for instance, when you need to shorten attribute names. For example, EstimatedTimeOfArrival can be abbreviated to ETA.
To edit a parameter and associate it with an input column:
1. In the Parse Expression pane, edit the parameter name as required.
2. From the Attribute drop-down list, select the desired input column.

Testing Expressions
You test your expression to check that results are as expected. The following figure is an example of an expression that has been evaluated and tested.

**Note** Testing an expression that contains an analytic function will validate the syntax without actually executing the function. Additionally, the test will only be performed on a single record.

**Note** Compose does not check the data types of columns used in an expression for compatibility. For example, if a column of type integer is used in an expression for a column of type varchar, the expression will not be executed successfully.

Figure 7.2 | Test Expression

![Test Expression Diagram]
To test an expression:

1. In the Expression Builder window, build an expression as described in Building Expressions.
2. Click Parse Expression as described in Parsing Expressions.
3. View the parameters that are displayed. If your expression is not valid, an error message is displayed.
4. Optionally edit the parameters name(s) as described in Editing Parameter Names.
5. Type values for each parameter and then click Test Expression to see the expression result.
   For example, using the expression in Test Expression, type Mike for FirstName and Smith for LastName. The result displayed is Mike Smith.
6. This step is only available for transformations created in the Edit Mappings dialog box. When you create a transformation in the Edit Mappings dialog box, an additional button called Show Data appears to the left of the Test Expression button. You can click this button to see how your expression translates into actual data.
   For example, clicking the Show Data button for the expression UnitPrice*Quantity will open the following window.
For more information on the **Edit Mappings** dialog box, see *Editing Column Mappings* in *Creating and Managing the Data Warehouse*.

**Defining Reusable Transformations**

In a single Compose project there may be several processes that require similar data transformations. For example a reusable transformation can be defined that concatenates first and last names. This transformation could then be used both in the Customers mapping and in the Employees mapping.

As opposed to stored functions or procedures which are environment dependent, reusable transformations are environment agnostic, meaning that not only can they be used as required within a Compose project, but they can also be used across different environments (using Compose’s export/import function).
Centrally managed transformations increase efficiency by eliminating unnecessary duplication, while at the same time, enabling the seamless propagation of changes to all transformation instances.

**To define a reusable transformation:**

1. From the drop-down menu in the top right of the Model panel, select Reusable Transformations.
   
   The Reusable Transformations window opens.
   
   The window is split into the following panes:
   
   - Upper pane - Lists the reusable transformations that have been defined.
   - Lower pane - Provides additional information about transformation instances such as where they are in use (e.g. mappings, model, etc.) and the expression that was created using the transformation.

   Select a transformation to see the additional information.

2. Click the New Transformation toolbar button.
   
   The New Transformation dialog box opens.
   
   a. In the Name field, specify a name for the transformation.
   
   b. In the Category field, specify a category name. If the category name already exists it will be displayed below the field when you start to type the name. To group the new transformation in the same category, simply select the existing name (unless of course you wish to create a new category with a similar name).

   In the Expression Builder, transformations are grouped according to their category name, making it easier to find the transformation you want to use. Therefore, when specifying a category name, it is recommended to choose a name that reflects the purpose of the transformation. For example, if you create several transformations that concatenate data, it would make sense to group those transformations under a category called "Join".
   
   c. To add a parameter to the transformation, click the New button to the right of the Parameters heading.

   A new row is added to the Parameters list.

   d. Specify a name for the parameter, select an appropriate data type, and optionally provide a description.

   Note  If you add multiple parameters, you can change a parameter’s position by selecting the parameter and then using the Up/Down arrows (above the Parameters list) to reposition it.

   e. Click the Create Expression button below the Parameters list.

   The Edit Transformation dialog box opens.

   f. In the Edit Transformation dialog box, create an expression using the parameters you defined earlier.
For information on creating expressions, see Creating Expressions.

g. Click OK to save the transformation. The transformation is added to the list in the upper pane. Once a transformation has been defined, it will be available for selection as needed in the Expression Builder’s Transformations tab. For information on creating expressions, see Creating Expressions.

Managing Reusable Transformations
You can manage reusable transformation as described in the table below.

<table>
<thead>
<tr>
<th>Table 7.5</th>
<th>Lineage Options</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>To</strong></td>
<td><strong>Do This</strong></td>
</tr>
<tr>
<td>Delete a transformation</td>
<td>Select the transformation and then click the <strong>Delete</strong> toolbar button. When prompted to confirm the action, click <strong>OK</strong>.</td>
</tr>
<tr>
<td><strong>Note</strong></td>
<td>If the transformation is in use, you first need to delete the transformation instances.</td>
</tr>
<tr>
<td>Edit a transformation</td>
<td>Double-click the transformation or select the transformation and then click the <strong>Edit</strong> toolbar button. Continue as described in Defining Reusable Transformations.</td>
</tr>
<tr>
<td><strong>Note</strong></td>
<td>Any changes you make to a transformation will be propagated to all instances of that transformation.</td>
</tr>
<tr>
<td>Edit a parameter</td>
<td>Open the <strong>Edit Transformation</strong> dialog box as described in Defining Reusable Transformations. Then, select the parameter you want to delete and click the <strong>Delete</strong> button above the Parameters list.</td>
</tr>
</tbody>
</table>
8 | Creating and Managing Command Tasks

Command tasks enable you to incorporate custom processes into your Compose workflow. This is especially useful if you need to leverage external tools to transfer files, validate data, and so on. A Command task can run any script or executable supported by the operating system including batch files, Python scripts, PowerShell scripts, executables and so on.

In this chapter:
Defining Command Tasks
Managing Command Tasks
Controlling and Monitoring Command Tasks

Defining Command Tasks
This section explains how to define a command task. You can define as many command tasks as you need and execute them at different stages of a Compose workflow.

Note Before you define a command task, make sure that the executable or script file that you want to run resides in the following directory on the Compose server machine:
PRODUCT_DIR\data\projects\YOUR_PROJECT\scripts

To define a command task:
1. From the project drop-down menu, select Manage Command Tasks.
   The Manage Command Tasks dialog box opens.
2. Provide a name for the task.
3. Optionally, enter a description.
4. In the Script/Executable File field, specify the name of the files that you want to run.
5. In the Parameters field, specify any parameters required by the command. Parameters should be separated by a space.
6. The user context is the user account under which the Task will run. To change the current user context, provide the User, Password and Domain of the account under which you want the Task to run.
7. Click **Save** to save your changes or **Discard** to discard any unsaved changes. The task will be added to the list of tasks in the left of the dialog box.

### Managing Command Tasks

The table below describes the task management options.

<table>
<thead>
<tr>
<th>To</th>
<th>Do This</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit a task</td>
<td>Select the task in the tasks list in the left of the Manage Command Tasks dialog box and edit it as described in Defining Command Tasks.</td>
</tr>
<tr>
<td>Delete a task</td>
<td>Select the task in the tasks list in the left of the Manage Command Tasks dialog box and then click the <strong>Delete</strong> toolbar button. When prompted to confirm the deletion, click <strong>OK</strong>.</td>
</tr>
<tr>
<td>Search for a task</td>
<td>Enter part of the task name in the search box above the task list. The list of tasks will be filtered to show only tasks that include the search term in their name.</td>
</tr>
</tbody>
</table>

### Controlling and Monitoring Command Tasks

Command Tasks can be run from the Manage Command Tasks dialog box or from the main Compose Monitor view. Although they can be run individually, command tasks are usually run as part of a workflow.

For information on defining workflows, controlling and monitoring tasks, and controlling and monitoring workflows, see  [Controlling and Monitoring Tasks and Workflows](#).

**To run a command task from the Manage Command Tasks dialog box:**

1. Open the Manage Command Tasks dialog box and select the task you want to run.
2. Click the **Run** toolbar button.
3. The Manage Command Tasks dialog box switches to Monitor view.
   - In Monitor view the following information is available:
     - The task ID
     - The current status
     - When the task started and ended
     - The overall task progress
9 | Creating and Managing the Data Warehouse

Once your model is set up properly, the next step in the Compose workflow is to create the Data Warehouse tables, generate the ETL set(s) and run the data warehouse ETL task.

In this chapter:
- Data Warehouse ETL Tasks
- Managing ETL Sets
- Viewing and Exporting ETL Commands
- Modifying ETL Settings
- Validating the Data Warehouse

Data Warehouse ETL Tasks

This section describes how to create the data warehouse tables, generate the ETL Set and run a data warehouse ETL task. It contains the following topics:

- How Compose Handles Missing References in the Data Warehouse
- Creating the Data Warehouse Tables
- Generating the ETL Set
- Controlling Data Warehouse ETL Tasks

How Compose Handles Missing References in the Data Warehouse

Before running a data warehouse task, it is important to understand how Compose handles missing references. Missing references may involve records that are simply missing or records whose arrival has been delayed. The latter might occur if data is ingested from two different systems (for example, an ERP system and a CRM system), with each system having its own ETL set.

» Handling an early-arriving fact - If a record references another record which does not exist yet, then Compose will do the following:

» Insert a placeholder for the missing reference record. The placeholder record will only include the business key and surrogate key. The rest of the columns will be set to NULL.
Document the missing record in the `TLOG_REF_ERRORS_VALUES` table. The `TLOG_REF_ERRORS_VALUES` table contains the following columns:

- **RUNNO** - The ETL task run number.
- **RELATIONNR** - An internal number that can be used by Attunity Support to determine the source entity.
- **NO_RELATIONS** - The number of missing references. For example, if Customer A ordered three different items (from the Orders table) and Customer A is missing, this number will be three.
- **KEYVALUE1-20** - The missing record. Since the missing record is a Primary Key, which may consist of several columns, there are 20 `KEYVALUE` columns.

**Example:**
If the "Orders" table references "SuperGlue" in the "Products" table, but "SuperGlue" does not exist in that table, Compose will mark "SuperGlue" as a missing reference, insert a record with the key value "SuperGlue" (assuming that the product name is the business key) to the "Products" table, and insert NULL values in the remaining "Products" table columns.

When the missing reference eventually arrives, it will be mapped to the record created for it and the NULL values will be replaced by the actual values.

**Note** If the record is defined as history type 2, the record with the NULL values will remain as a historical record.

See also: Viewing Missing References.

Creating the Data Warehouse Tables
When you create the data warehouse tables, two types of tables are created: staging tables (indicated by the TSTG prefix) and the actual data warehouse tables (indicated by the TDWH prefix).

Compose adds `RUNNO_INSERT` and `RUNNO_UPDATE` columns to both the data warehouse tables and the data mart tables. These columns contain the ETL task run number, which can be used (in the Run Details window or in Details tab) to find out more information about the task (e.g. the number of rows updated or inserted per table). Note that in hub tables and type 1 dimensions, the `RUNNO_UPDATE` number will usually be higher than the `RUNNO_INSERT` number as these tables do not contain any history. In satellite tables or type 2 dimension tables however, the `RUNNO_INSERT` number and the `RUNNO_UPDATE` number will always be the same as a new row is inserted for each update (i.e. history is retained).
To create the data warehouse tables:

1. Click the **Create** button in the bottom right of the **Data Warehouse** panel. The **Creating Data Warehouse** window opens. A progress bar indicates the current progress. For each stage of the Data Warehouse generation process, a corresponding message appears in the **Messages** list.

   ![Creating Data Warehouse](image)

   **Note**  When creating table in a Microsoft SQL Server data warehouse, you may encounter the following error:

   Data warehouse creation failed. Error: Cannot create a row of size 11272 which is greater than the allowable maximum row size of 8060. The statement has been terminated.

   This is a well-documented Microsoft SQL Server limitation. To work around this limitation you need to split the offending table(s) into smaller tables.

2. When the "Data warehouse created successfully" message appears, click **Close**.

**Viewing the Data Warehouse Tables**

After the data warehouse tables are created, you can view them by clicking the number to the left of the **Data Warehouse Tables Present** text in the Data Warehouse panel.

![Data Warehouse Tables Present](image)

When you click the link, the **Data Warehouse Tables** window opens showing a list of all the tables in your data warehouse.
Compose adds **RUNNO_INSERT** and **RUNNO_UPDATE** columns to both the data warehouse tables and the data mart tables. These columns contain the ETL task run number, which can be used (in the Run Details window or in Details tab) to find out more information about the task (e.g. the number of rows updated or inserted per table). Note that in hub tables and type 1 dimensions, the **RUNNO_UPDATE** number will usually be higher than the **RUNNO_INSERT** number as these tables do not contain any history. In satellite tables or type 2 dimension tables however, the **RUNNO_INSERT** number and the **RUNNO_UPDATE** number will always be the same as a new row is inserted for each update (i.e. history is retained).

**Note**  In the data mart tables, the **RUNNO_INSERT/RUNNO_UPDATE** column names are prefixed by the table name e.g. **ORDERS_RUNNO_UPDATE**.

To view a specific table, simply double-click the table.

**Note**  Apart from the **Date** and **Time** tables which are automatically populated on creation, the other tables will be empty until you run the data warehouse ETL task.

See [Controlling Data Warehouse ETL Tasks](#) below for information on running a data warehouse ETL task.

In the `<Table Name>` window, you can perform the following tasks:

- Choose how many rows to display from the **Rows** drop-down list.
- Click the **Column Settings** button to choose which columns will be displayed and the order in which they will be displayed.

**Generating the ETL Set**

After the data warehouse tables have been created, you then need to generate the ETL Set that will be used in the data warehouse ETL task. The ETL Set contains the Mappings ETL (which is automatically created) and any custom ETLs that you have defined. If you need to make changes to the Mappings or define custom ETLs, continue from [Managing ETL Sets](#) and [Creating Custom ETL Scripts](#) respectively.

To generate the data warehouse ETL set:

1. Click the **Manage** button in the bottom left of the **Data Warehouse** panel. The Manage ETL Sets window opens.
2. If you have more than one ETL Set, in the left pane, select the ETL Set that you want to generate.
3. Click the **Generate** toolbar button. The **Generating ETL Set for Name** progress window opens. When the "Generate ETL
finished successfully" message is displayed, close the window.

**Note** Only mappings selected in the Manage ETL Sets window will be generated.

Controlling Data Warehouse ETL Tasks

Once the data warehouse tables have been created and the ETL Set has been generated, you can then proceed to run the data warehouse ETL task. The data warehouse ETL task extracts data from the staging tables and load it into the data warehouse tables.

**Note** Ingesting a historical record deletes any history that is later than the ingested record. For example, let's assume that data warehouse contains the following historical records:

- 2012 - Boston
- 2014 - Chicago
- 2015 - New Jersey

Ingesting the record 2013 - New York will delete the 2014 and 2015 records.

Data warehouse ETL tasks can be run manually, scheduled to run periodically or run as part of a workflow. The section below describes how to run a data warehouse task manually. For information on scheduling data warehouse tasks or including them in a workflow, see Controlling and Monitoring Tasks and Workflows.

**To run a data warehouse ETL task:**

1. Click the Manage button in the bottom right of the Data Warehouse panel. The Manage ETL Set dialog box opens.
2. If you have more than one ETL Set, in the left pane, select the ETL Set that you want to generate.
3. Click the Run toolbar button. The dialog box switches to Monitor view and a progress bar shows the current progress in terms of percentage.

   When the progress reaches 100% completed, close the Manage ETL Set dialog box.

   You can stop the task at any time by clicking the Abort toolbar button. This may be necessary if you need to urgently edit the task settings due to some unforeseen development. After editing the task settings, simply click the Run button again to restart the task.

   **Note** Aborting a task may leave the data warehouse tables in an inconsistent state. Consistency will be restored the next time the task is run.
Other monitoring information such as the task details (i.e. the number of rows inserted/updated) and the task log files can be accessed by clicking the **Run Details** and **Log** buttons respectively.

Once the data warehouse has been successfully loaded into the data warehouse tables, you can then proceed to the final part of the Compose workflow - defining and populating data marts. For more information, see *Creating and Managing Data Marts*.

### Managing ETL Sets

An ETL Set contains the mappings between the columns in the landing area tables and the columns in the staging tables. The same mappings can be used by several ETL Sets. You can create new ETL Sets, duplicate ETL sets and edit existing ETL Sets as required.

The following options are available:

- Adding and Duplicating ETL Sets
- Editing Column Mappings
- Creating Custom ETL Scripts

**Note** You must regenerate the ETL Set and then run a data warehouse ETL task whenever the mappings are modified or whenever custom ETLs are added or modified. Populating the data warehouse can either be done manually as described in *Controlling Data Warehouse ETL Tasks* or automatically as described in *Scheduling Tasks*. If you have already run the data mart ETL tasks, then you also need to regenerate the data mart ETLs and run the tasks again as described in *Creating and Managing Data Marts*.

### Adding and Duplicating ETL Sets

As the default ETL Sets are generated automatically, there is usually no reason to manually create or duplicate an ETL Set. An exception to this is if you import your model from ERwin without first defining global mappings. In such a case, you will need to manually add the ETL Set and create the mappings.

For more information on global mappings, see *Managing Global Mappings*.

One possible reason to duplicate an ETL set is if your model contains different types of tables and you want to manage them in separate ETLs.

### Table 9.1 | Mapping Editing Options

<table>
<thead>
<tr>
<th>To Add a new ETL Set</th>
<th>Do This</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Click the <strong>Manage</strong> button at the bottom left of the <strong>Data Warehouse</strong> panel.</td>
<td></td>
</tr>
</tbody>
</table>
Table 9.1 | Mapping Editing Options

<table>
<thead>
<tr>
<th>To</th>
<th>Do This</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The Manage ETL Set dialog box opens.</td>
</tr>
<tr>
<td>2.</td>
<td>Click the New toolbar button.</td>
</tr>
<tr>
<td></td>
<td>The Add New ETL Set dialog box opens.</td>
</tr>
<tr>
<td>3.</td>
<td>Specify a name for the ETL Set and then click OK.</td>
</tr>
<tr>
<td></td>
<td>The ETL Set added.</td>
</tr>
<tr>
<td>4.</td>
<td>Select the ETL Set name in the left pane and continue from Editing</td>
</tr>
<tr>
<td></td>
<td>Column Mappings.</td>
</tr>
</tbody>
</table>

Duplicate an ETL Set

| 1.          | Click the Manage button at the bottom left of the Data Warehouse       |
|             | panel.                                                                 |
|             | The Manage ETL Set dialog box opens.                                   |
| 2.          | Select the ETL you want to duplicate and then click the Duplicate      |
|             | toolbar button.                                                       |
|             | The Duplicate dialog box opens.                                        |
| 3.          | Specify a Name for the new ETL Set.                                    |
| 4.          | Select a Landing Area.                                                |
| 5.          | Optionally change the default Schema.                                  |
| 6.          | Select an ETL Type according to your Replicate task type.              |
| 7.          | Click OK.                                                              |
| 8.          | Select the ETL Set name in the left pane and continue from Editing      |
|             | Column Mappings.                                                       |

Editing Column Mappings

The mappings show the current mapping between the landing area tables and the staging tables. By default, the columns names and data in the source tables and the staging tables will be identical. However, you can manually change the mappings according to your needs, either by simply mapping a source column to a different staging column and/or by using an expression.

**To edit column mappings:**

1. Click the Manage button in the Data Warehouse panel.
2. In the Manage ETL Set dialog box, select the Mappings tab.
   - Each of the staging tables has a corresponding mapping name.
3. In the **Mappings** column, click the mapping that you want to edit. The **Edit Mapping: Name** window opens.

4. Edit the mapping as described in the table below.

### Table 9.2 | Mapping Editing Options

<table>
<thead>
<tr>
<th>To</th>
<th>Do This</th>
</tr>
</thead>
<tbody>
<tr>
<td>Map a column in a landing area</td>
<td>Note: The mapping procedure differs depending on whether you are in</td>
</tr>
<tr>
<td>area table to a column in a</td>
<td>Standard View or Compact View. For information on changing the view, see</td>
</tr>
<tr>
<td>staging area table</td>
<td>Change the view.</td>
</tr>
</tbody>
</table>

**In Standard View:**

1. Hover the mouse cursor over the source column name as shown in the image below. A gray dot appears to the right of the column name.

2. Drag the mouse cursor from the gray dot to the desired column in the staging table.

3. When the dotted line turns green (as shown below), release your mouse button. The mapping operation is completed.

   Note that if the dotted line turns red (instead of a green), you will not be able to map the source column with the desired staging
Table 9.2 | Mapping Editing Options

<table>
<thead>
<tr>
<th>To</th>
<th>Do This</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>column. A red dotted line indicates that the source and staging column data types are incompatible with each other.</td>
</tr>
<tr>
<td></td>
<td><strong>In Compact View:</strong></td>
</tr>
<tr>
<td></td>
<td>1. Switch to Compact View as described in Change the view.</td>
</tr>
<tr>
<td></td>
<td>2. Drag the source column to the cell located to the left of the target staging column.</td>
</tr>
<tr>
<td>Auto-generate mapping</td>
<td>Click the <strong>Auto-Map</strong> toolbar button.</td>
</tr>
<tr>
<td>Remove all mappings</td>
<td>Click the <strong>Reset</strong> toolbar button.</td>
</tr>
<tr>
<td>Change the view</td>
<td>Changing to a more compact view is recommended for sources tables that have numerous columns. In compact view, the table columns are organized in rows (instead of a single list), making it easier to locate source columns and map them to the desired staging columns. You can also use the search box to filter out all columns that do not match the search string.</td>
</tr>
<tr>
<td></td>
<td><strong>To change the view:</strong></td>
</tr>
<tr>
<td></td>
<td>Click the <strong>Change View</strong> toolbar button. For information on creating mappings in Compact view, see Map a column in a landing area table to a column in a staging area table.</td>
</tr>
<tr>
<td>Select a different source database</td>
<td>Select a database from the <strong>Landing Area Database</strong> drop-down list on the left of the dialog box.</td>
</tr>
<tr>
<td>Select a different source schema</td>
<td>Select a schema from the <strong>Schema</strong> drop-down list on the left of the dialog box.</td>
</tr>
<tr>
<td>Change the entity type</td>
<td>Select <strong>Table</strong>, <strong>View</strong> or <strong>Query</strong> on the left of the dialog box. If you choose the <strong>Query</strong> option, see also Define a custom query.</td>
</tr>
<tr>
<td>Define a custom query</td>
<td>When the entity type is set to <strong>Query</strong>. You can set a custom select query instead of using the existing source tables/views.</td>
</tr>
<tr>
<td></td>
<td><strong>To set a query:</strong></td>
</tr>
<tr>
<td></td>
<td>1. Click the <strong>Set Query</strong> button.</td>
</tr>
<tr>
<td></td>
<td>The <strong>Edit Mapping Select Query: &lt;Mapping Name&gt;</strong> dialog box opens.</td>
</tr>
</tbody>
</table>
| | Hover the mouse cursor over a table and/or a column and then click.
To Do This
---
- Click the arrow to the right of the highlighted table/column to add it to the Query.
- Use the **Select** button at the top of the dialog box to add select statements to your query.
- Optionally use the **Undo**, **Redo** and **Clear** buttons as required.
  1. Click **OK** to save your settings and close the dialog box.

2. **Click OK** to save your settings and close the dialog box.

The query results will be displayed on the left of the **Edit Mappings: <Name>** dialog box.

<table>
<thead>
<tr>
<th>Select a different table</th>
<th>Select a table from the <strong>Table Name</strong> drop down list on the left of the dialog box.</th>
</tr>
</thead>
<tbody>
<tr>
<td>See the data of a selected table</td>
<td>Select a source table and then click the <strong>Show Source Data</strong> button on the left of the dialog box.</td>
</tr>
</tbody>
</table>

| Create a table-level transformation (Filter) | 1. Click the **Filter** toolbar button in the **Edit Mappings: Name** dialog box. The Expression Builder opens. 2. Continue from **Opening the Expression Builder.** |

**Note** When creating a filter for a table, the expression should return 1 for data that you want to include and 0 for data that you want to exclude.

**Note** The assumption is that columns that are used in a filter do not change between different versions of the record. Filtered out records are lost forever, so if you also need to retain the historical records do not filter them out.

| Create a column-level transformation | 1. Hover the mouse cursor over the Staging Column for which you want to create a transformation and then click the $fx$ button that appears to its right. The Expression Builder opens. 2. Continue from **Opening the Expression Builder.** |

---

Table 9.2 | Mapping Editing Options
---
<table>
<thead>
<tr>
<th>To</th>
<th>Do This</th>
</tr>
</thead>
<tbody>
<tr>
<td>Select a different table</td>
<td>Select a table from the <strong>Table Name</strong> drop down list on the left of the dialog box.</td>
</tr>
<tr>
<td>See the data of a selected table</td>
<td>Select a source table and then click the <strong>Show Source Data</strong> button on the left of the dialog box.</td>
</tr>
<tr>
<td>Create a table-level transformation (Filter)</td>
<td>1. Click the <strong>Filter</strong> toolbar button in the <strong>Edit Mappings: Name</strong> dialog box. The Expression Builder opens. 2. Continue from <strong>Opening the Expression Builder.</strong></td>
</tr>
</tbody>
</table>

**Note** When creating a filter for a table, the expression should return 1 for data that you want to include and 0 for data that you want to exclude.

**Note** The assumption is that columns that are used in a filter do not change between different versions of the record. Filtered out records are lost forever, so if you also need to retain the historical records do not filter them out.

| Create a column-level transformation | 1. Hover the mouse cursor over the Staging Column for which you want to create a transformation and then click the $fx$ button that appears to its right. The Expression Builder opens. 2. Continue from **Opening the Expression Builder.** |
Adding, Deleting and Renaming Mappings
You can add, rename and delete mappings as required. For example, if you want one of the staging tables to contain columns from several tables in the landing area, then you need to add a new mapping for each of the landing area tables.

To add, delete, and rename mappings:
1. Click the Manage button in the Data Warehouse panel. The Manage ETL Set dialog box opens.
2. In the left pane, select the ETL Set you want to add, delete, or rename.
3. Select the Mappings tab.
4. Add or delete mappings as described in the following table.

<table>
<thead>
<tr>
<th>To</th>
<th>Do This</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add a new mapping</td>
<td>1. In the Staging Tables column, select the staging table that you want to map.</td>
</tr>
<tr>
<td></td>
<td>2. Click the New button above the Staging Tables column.</td>
</tr>
<tr>
<td></td>
<td>The New Mapping dialog box opens.</td>
</tr>
<tr>
<td></td>
<td>3. Optionally change the default mapping name.</td>
</tr>
<tr>
<td></td>
<td>4. Click OK to save the mapping.</td>
</tr>
<tr>
<td></td>
<td>5. Enable the mapping.</td>
</tr>
</tbody>
</table>

| Delete a mapping    | 1. In the Mappings column, hover the mouse cursor over the mapping you want to delete. |
|                     | 2. Click the Delete(x) button that appears to its right.                   |
|                     | 3. Click OK when prompted to confirm the deletion.                         |

| Rename a mapping    | 1. In the Mappings column, hover the mouse cursor over the mapping you want to rename. |
|                     | 2. Click the Rename (A) button that appears to its right.                  |
|                     | The Rename dialog box opens.                                              |
|                     | 3. Specify a new name for the mapping and then click OK.                   |

Handling Duplicate Business Keys
When two or more records in the data source have the same business key, you can select the Handle Duplicates check box to prevent an error from occurring when the data warehouse ETL task is run. When this check box is selected, Compose will only add one of the records to the data warehouse.
Note  Since Compose randomly chooses which record to add to the data warehouse, you may want to run a data warehouse ETL task first to see if there are any duplicate record errors. In the event that there are, you can then modify the data source to remove records that have the same business key.

You should also select the Handle Duplicates check box in the following situations:

- The Data Warehouse ETL type is either Full Load and Change Tables or Change Tables Only.
  This is because the Change Tables may contain two records with the same business key: The old record and the updated record. When the Handle Duplicates check box is selected, the updated record will always be inserted/updated to/in the data warehouse.

- When a single table in the data warehouse is derived from multiple landing area tables, the same business key will be set for each of the mappings. To prevent an error for occurring, you need to select the Handle Duplicates check box.

Handling Null Updates
The default handling of null updates is set in the Advanced tab of the Settings - <ETL_Set_Name> window. For each mapping, you can override the specified default.

To do this
1. In the Manage ETL Sets window, select the desired mapping.
2. Click the Null Updates toolbar button.
3. Select one of the available options.
   For a description of the options, see Handling Null Updates.

Using Lookup Tables
Lookup tables are useful for replacing source data with the actual data that you want to appear in the data warehouse. For example, a lookup table could be used to replace a zip code with a full address or, conversely, to replace a full address with a zip code.

To link a lookup table column to a staging table column:
1. Click the link to the desired ETL set in the Data Warehouse panel.
   The Manage ETL Set dialog box opens.
2. In the Mappings column, click the mapping for the staging table containing the result column (with the data that you want to replace).
   The Edit Mapping - Name dialog box opens.
3. Hover the mouse cursor over the relevant staging column and then click the Lookup button that appears to the right of the column name.
The **Select Lookup Table** dialog box opens.

a. From the **Database** drop-down list, select the database containing the lookup table.

   **Note** The database must reside in your data warehouse.

b. From the **Schema** drop-down list, select the schema containing your source lookup tables.

c. Select either **Table** or **View** according to the lookup table type.

d. From the **Table** drop-down list, select the lookup table.

   The right side of the **Select Lookup Table** dialog box displays the lookup table columns and their data types. To view the data in the lookup table, click the **Show Lookup Data** button.

e. After you have selected the lookup table, click **OK**.

   The **Lookup Transformations - Table Name.Column Name** dialog box opens.

   The dialog box is divided into the following panes:

   - **Upper pane**: The upper part of the right pane (**Condition**) displays the condition expression, which stipulates the condition(s) for performing the lookup.
   - **Lower pane**: The lower part of the right pane (**Result Column**) displays the column result expression, which stipulates what data to replace in the target column.

4. To change the lookup table, click the **Change Lookup Table** button above the lookup table columns and then perform steps a. to d. above.

5. To view the lookup table or landing table data, click the **Show Lookup Data** or **Show Landing Data** buttons respectively.

6. To specify the condition(s) for performing the lookup, click the **Create Expression** button (which changes to **Edit Expression** after an expression has been created) above the **Condition** expression.

   The **Condition Expression - Column Name** dialog box opens.

7. Create an expression using the landing and lookup table columns on the left.

   For an example, see **Lookup Example**.

   For information on creating expressions, see **Creating Expressions**.

8. To specify what data to replace or add if the lookup conditions are met, click the **Create Expression** button (which changes to **Edit Expression** after an expression has been created) above the **Result Column** expression.

   The **Result Expression - Column Name** dialog box opens.

9. Create an expression using the landing and lookup table columns on the left.

   For an example, see **Lookup Example**.

   For information on creating expressions, see **Creating Expressions**.

10. To preview the results, click the **Preview Results** button.

11. Click **OK** to save your settings and close the **Lookup Transformations - Table Name.Column Name** dialog box.
Using Lookup Tables that do not have an ETL Set for CDC Mapping

When the **Store Changes** option is enabled in the Replicate task, Replicate creates Change Tables in the landing area. These tables contain only the changes to the original data. The Compose ETL Set CDC task reads the changes from Change Tables and applies them to the target tables. However, if the landing area contains dedicated lookup tables (i.e. tables that are not associated with any Compose task), Compose will not be able to apply changes to these tables.

There are two ways of handling such a scenario, both of which are described below.

**Method 1**
Define another Replicate task with the **Apply Changes** replication option enabled.

**Method 2**

1. **Discover the landing site** and add all the lookup tables to the Compose model without any relation to/from other entities.
2. Either, **define lookups** from the data warehouse hub tables to the newly added entities.
   - **OR-**
     **Create relationships** from the data warehouse hub tables to the newly added entities.

   **Note** Creating relationships may not be a viable option when the lookup tables are complex.

3. Define a new DWH ETL **Change Tables Only** task that updates the lookup tables.
4. Ensure that the new task runs before the DWH ETL task.

The advantage of this method is twofold: a.) All the tables used in the mappings are managed by Compose, and b.) Only one Replicate task needs to be defined (which also means that the database transaction logs are read only once). The disadvantage is that you need to ensure that the ETL task that updates the lookup entities always runs before any data warehouse ETL task.

**Lookup Example**
The following example shows how a lookup table is used to concatenate a Dutch translation of the category name (located in the lookup table) to the original category name located in the landing table.

The lookup could be defined using the following expressions:

1. **Condition expression:**
   
   ```
   ${Lookup.CategoryID}=${Landing.CategoryID}
   ```

   **Meaning:** Perform the lookup only if the Category ID in the landing table and the lookup table are the same.

2. **Result column expression:**
${Lookup.CategoryName} + 'is' + ${Landing.CategoryName}

**Meaning:** Add the data in the *CategoryName* column in the lookup table to the data in the *CategoryName* column in the landing table (separated by the word "is").

Assuming the result column name is "Split Name", clicking the *Preview Results* button would display the following table:

<table>
<thead>
<tr>
<th>Split Name</th>
<th>Category Name (Lookup)</th>
<th>Category Name (Landing)</th>
<th>Category ID (Lookup)</th>
<th>Category ID (Landing)</th>
</tr>
</thead>
<tbody>
<tr>
<td>dranken is Beverages</td>
<td>dranken</td>
<td>Beverages</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Specerijen is Condiments</td>
<td>Specerijen</td>
<td>Condiments</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Gebak is Confectionary</td>
<td>Gebak</td>
<td>Confectionary</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Zuivelproducten is Dairy Products</td>
<td>Zuivelproducten</td>
<td>Dairy Products</td>
<td>4</td>
<td>4</td>
</tr>
<tr>
<td>Grains/Granen is Grains/Cereal</td>
<td>Grains/Granen</td>
<td>Grains/Cereal</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>Vlees/Gevolgete is Meat/Poultry</td>
<td>Vlees/Gevolgete</td>
<td>Meat/Poultry</td>
<td>6</td>
<td>6</td>
</tr>
</tbody>
</table>

**Creating Custom ETL Scripts**

In addition to the *Mappings* ETL, you can define custom ETLs as required. User-defined ETLs can define specific transformations, cleansing and filtering processes.

**To create a custom ETL:**

1. Click the *Manage* button in the bottom left of the *Data Warehouse* panel. The *Manage ETL Set* dialog box opens.
2. Select one of the following tabs according to your needs:
   - **Multi Table ETL** - to define an ETL for multiple tables.
   - **Single Table ETL** - to define an ETL for a single table.
   - **Post Loading ETL** - to define an ETL that will be executed after the data has been loaded from the staging tables to the data warehouse.
3. If you selected **Single Table ETL**, select an entity in the *Entity* column and then click the *New* button above the Entity list. For **Multi Table** and **Post Loading** ETLs, just click the *New* button.
4. Specify a name for your ETL and then click *OK.*
If you selected **Single Table ETL**, the ETL is added as a link to the **User Defined ETL** column. If you selected **Multi Table ETL** or **Post Loading ETL**, the ETL is added as a link in their respective tabs.

5. Click the link to open the **Edit ETL Instructions** dialog box.

6. If you selected **Single Table ETL**, select a column and click the arrow to the right of the selected column to add it to the ETL.

   If you selected **Multi Table ETL** or **Post Loading ETL**, select a table and a column and then click the arrow to the right of the selected table/column to add it to the ETL. Repeat as necessary.

7. Use the **Select, Delete, Insert** and **Update** toolbar buttons at the top of the dialog box to add SQL statements to your ETL.

8. To run the ETL as a stored procedure (that already exists in the data warehouse):
   a. Select the **Execute as Stored Procedure** check box.
   b. Click the **Stored Procedure** toolbar button.
   c. Replace **STORED_PROCEDURE** with the name of your stored procedure and replace (PARAM1, PARAM2) with any parameters that it needs. Note that parameters must be separated by a comma. If no parameters are required, use empty parenthesis or drop them altogether.

   **Note** By default, Compose connects to Teradata in ANSI mode. To run a Teradata stored procedure that was created in **TERA** (Teradata) mode, you also need to add the parameter **TMODE=TERA** to the **Additional JDBC Parameters: field in the Teradata Data Warehouse connection settings**.

8. Use the Undo, Redo and Reset buttons at the bottom of the window as needed.

9. Optionally, specify a description in the **Description** box at the bottom of the window.

10. To save your ETL, click **OK**.

**Single Table Example**

The following example, based on the **Attunity Compose Tutorial** in **Getting Started with Attunity Compose**, demonstrates how to concatenate two columns called "First Name" and "Last Name" into a single column called "FullName".

1. Click the **Manage** button in the **Model** panel.
   The **Manage Model** window opens.

2. Select **Employees** from the **Entities** list on the left.

3. Click the + (plus) toolbar button to add a new **Attribute**.
   A new row is added to the **Attributes** table.

4. Type any letter in the **Column Name** column to bring up the "Add New" option. Click the "Add New" option when it appears.
The New Attribute Domain window opens.

5. In the Name field, type FullName. From the Type drop-down list, select Varchar. In the Length field, enter 100.

6. In the History column, select Type 1 from the drop-down list.

7. Click OK to close the New Attribute Domain window and add the attribute to the Attributes table.

8. Then click OK again (below the newly added attribute) to exit edit mode.


10. In the Data Warehouse panel, click the Create button.

11. After the Data Warehouse tables have been created, close the Creating Data Warehouse window.

12. Click the Manage button in the bottom left corner of the Data Warehouse panel.

   The Manage ETL Sets window opens.

13. To view the current mappings between the source columns and staging columns, click the Map_Employees_1 link in the Mappings column.

   A "Processing" icon is displayed while the mappings are generated. After the mappings are generated, the Edit Mappings - Map_Employees_1 window opens automatically.

   You will notice that the FullName column has been added to the staging columns, but is currently not mapped to the source columns.
14. The next stage is to define an ETL that will map the **First Name** and **Last Name** source columns to the **Full Name** staging column.

15. Close the **Edit Mappings - Map_Employees_1** window and then select the **Single Table ETL** tab on the left.

16. Select **Employees** in the **Entity** column and then click the **New** button above the column. The **Add New Single Table ETL** dialog box opens.

17. Specify a name or leave the default name and then click **OK**.

18. Click the **Edit** button (represented by a pencil icon) at the end of the **Employees** row. The **Edit Single Table ETL: <Name>** window opens.

19. In the editing pane on the right, enter the following instruction:

   ```sql
   UPDATE dbo.TSTG_EMPLOYEES set
   FullName = LASTNAME + FIRSTNAME
   ```

20. Click **OK** to save the ETL and close the window.

**Note** After Compose has finished populating the Data Warehouse, you can open the table in Microsoft SQL Server Management Studio and verify that the new column has been added with the correct data.

### Dropping and Recreating Tables

You can drop and recreate tables in your data warehouse as required. If you change the model after the data warehouse tables have already been created and loaded with data, you should adjust the data warehouse to reflect the modified model (as described in **Validating the Data Warehouse**). Some changes however cannot be resolved by adjusting the data warehouse. In such cases, you can either revert the model to its pre-modified state or drop and (optionally) recreate the data warehouse tables.

Note that dropping and recreating tables will delete all of the data in the tables and should only be performed in lieu of a better option.
Note In some scenarios, you need to edit the CREATE table statements before they are run. This can be done using the Generate DDL scripts but do not run them in Editing the Project Settings. For example, if your data warehouse tables contain partitions, you will need to edit the script to maintain the partitions. Another example is if you are using Teradata as your data warehouse and you want to change the Distribution Key set by Compose.

To drop and recreate tables:
1. In the Data Warehouse panel, select the Drop and Recreate Tables item from the menu in the top right corner.
   The Drop and Recreate Tables dialog box opens.
2. You can select to drop and/or recreate one or more of the following tables:
   » Data Warehouse & Data Marts - The data warehouse tables are derived from the model whereas the data mart tables are derived from the data warehouse tables.
   » ETL - These tables contain the ETL scripts. By default, these tables are prefixed with the string "TPIL".
   » Logging - These tables are generated when the ETL task runs and contain logging information. By default, these tables are prefixed with the string "TLOG".
   » Intermediate - These tables are temporary tables that are created when the ETL task runs. By default, these tables are prefixed with the string "TTMP".

Note Intermediate tables are created dynamically and therefore cannot be recreated.

3. Click OK to perform the drop and/or recreate operation.

Data Profiling
Data profiling is an analysis of the candidate data sources for a data warehouse to clarify the structure, content, relationships and derivation rules of the data. In short, data profiling helps you understand your data and model it correctly.

Attunity Compose enables you to profile the data in the landing area tables before it is loaded into the data warehouse. If you discover a problem with certain data, then you can either manually adjust the source tables or create a rule for handling the data in question.

To profile the data:
1. Click the Manage button at the bottom of the Data Warehouse panel.
2. In the Manage ETL Set dialog box, click the link in the Mapping column for the table you want to profile.
3. In the Edit Mappings - <Name> dialog box, click the Data Profiler toolbar button.
   The Profile <Table Name> (Landing Area) dialog box opens.
The following columns are displayed:

» **Column Name** - The name of the table column
» **Nulls** - The number of null values in the column
» **Count** - The number of rows in the column.
» **Count Distinct** - The number of unique rows in the column.
» **Duplicates** - The number of duplicate values in the column.

Note that although Compose calculates the number of duplicate values by subtracting Count Distinct from Count, the actual number of records displayed when you click the Duplicates number will be higher. This is because Compose has no way of knowing which of the records that share the same column value are legitimate duplicates (if any). It therefore displays all records that share the same value so you can decide which of them to delete (if any).

For example, in the Employees table, there may be several employees that live in London (the City column). Therefore duplicates of "London" are perfectly acceptable. However, two employees with the same phone number and a different address, for example, may indicate that the phone number in one of the records was entered incorrectly.

Duplicate values are quite common and usually do not indicate a problem. Where this feature is particularly useful however, is for detecting duplicate Primary Key candidate columns.

» **Data Type** - The column data type
» **Max** - The highest data value
» **Max Length** - The longest data value
» **Min** - The lowest data value
» **Min Length** - The shortest data value

4. For more information about a value, click the link in the column. A dialog box opens showing the record(s) containing the value. To add a Data Quality rule, click the Data Quality button and continue as described in Defining and Managing Data Quality Rules.

5. To only show columns that are mapped to a staging table column, select the Only show mapped columns check box.

6. To change the number or rows sampled, select a different value from the Rows to sample drop-down list. Note that the table may contain less rows than the selected value. The Sampled records value is the actual number of rows sampled.

7. To see all the table data, click the Show Data button.

8. To recalculate the data, click the Recalculate button. This is useful if the data in the landing area tables is being constantly updated (for example, due to a Replicate Change Processing task).

9. To search for a particular value, start typing the value in the Search box. Only values that match the search term will be shown.
Defining and Managing Data Quality Rules

There are many definitions of data quality but data is generally considered high quality if, "they are fit for their intended uses in operations, decision making and planning." (Tom Redman<Redman, T.C. (2008). With Compose, the data must be "fit" for use in a data mart.

Compose provides two ways of ensuring data quality: Data validation and data cleansing. As opposed to data validation which usually results in data being rejected, data cleansing provides a means of replacing, modifying, or deleting incomplete, incorrect or inaccurate data.

**Defining Data Cleansing Rules**

Attunity Compose enables you to define data cleansing rules for each of a table’s columns. Each rule consists of a data validation condition and a cleansing process that is performed as required (i.e. if the data is not valid).

**To add a rule:**

1. Click the **Manage** button at the bottom of the Data Warehouse panel.
2. In the **Manage ETL Set** dialog box, click the link in the **Mapping** column for the relevant table.
3. In the **Edit Mappings - <Name>** dialog box, click the **Data Quality** toolbar button.
   The **Data Quality Rules - <Table Name>** dialog box opens.
4. To add a new rule, click the **New** toolbar button.
   A row is added to the rules table.
5. In the **Name** column, specify a name for the rule.
6. From the drop-down list in the **Column** column, select the column to which the rule will be applied.
7. Hover the mouse-cursor over the **Condition** column and then click the fx button that appears on the right.
8. In the **Edit Condition Rule** dialog box, create a condition (using an expression) that the data in the column must meet in order to be considered valid. For more information on creating expressions, see **Opening the Expression Builder**.
   See also **Simple Example Rule** below.
9. From the drop-down list in the **If Condition is False** column, select **Cleanse Silently**.
10. Hover the mouse-cursor over the **Correction** column and then click the fx button that appears on the right.
11. In the **Edit Correction Rule** dialog box, create an expression to cleanse the data. For more information on creating expressions, see **Opening the Expression Builder**.
   See also **Simple Example Rule** below.
12. In the **Description** column, enter a description for the rule.
13. In the **Enabled** column, select or clear the check box to enable (the default) or disable the rule respectively.
Simple Example Rule

The condition expression on the left stipulates that the product ID number must be less than 100. If it is greater than or equal to 100, the data will be corrected using the expression on the right.

\[ \${\text{ProductID}} < 100 \] \[ \${\text{ProductID}} - 100 \]

**Defining Data Validation Rules**

Attunity Compose enables you to define data validation rules that are applied to the data before it is loaded into the data warehouse. In addition to defining rules, you can also define what action should be taken when data is rejected/accepted by Compose.

**To add a rule:**

1. Click the **Manage** button at the bottom of the Data Warehouse panel.
2. In the **Manage ETL Set** dialog box, click the link in the **Mapping** column for the table you want to profile.
3. In the **Edit Mappings - <Name>** dialog box, click the **Data Quality** toolbar button.
   - The **Data Quality Rules - <Table Name>** dialog box opens.
   - The default rule rejects primary keys that have a null value and reports the rows.
4. To add a new rule, click the **New** toolbar button.
   - A row is added to the rules table.
5. In the **Name** column, specify a name for the rule.
6. Hover the mouse-cursor over the **Rule** column and then click the **fx** button that appears on the right.
7. In the **Edit Data Quality Rule** dialog box create a rule using an expression. For more information on creating expressions, see **Opening the Expression Builder**.
   - See also **Simple Example Rule** below.
8. From the drop-down list in the **Error Action** column, select one of the following actions (performed when the data does not meet the rule conditions):

   - **Reject and report** - Reject the data and send a report
   - **Reject silently** - Reject the data without sending a report
   - **Reject and abort** - Reject the data and abort the data warehouse task
   - **Accept and report** - Accept the data and send a report
9. In the **Description** column, enter a description for the rule.
10. In the **Enabled** column, select or clear the check box to enable (the default) or disable the rule respectively.
Note: A rules that is defined to reject or accept a non-null value (e.g. 2) in a given column will also reject/accept NULL values that appear in the same column, but in different records. To prevent this from happening, add the following condition to the rule: "and column value is not null"

**Example:**

LEN(${CName}) < 2 and (${CName} is not null)

Simple Example Rule

The following rule stipulates that the number of units in stock must be greater than 1.

${UnitsInStock} > 1

**Managing Data Quality Rules**

The table below describes the available options for managing Data Quality rules.

<table>
<thead>
<tr>
<th>To</th>
<th>Do This</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit a Data Quality rule</td>
<td>Select the rule and edit it as described in Defining Data Cleansing Rules and Defining Data Validation Rules respectively.</td>
</tr>
<tr>
<td>Enable or disable a Data Quality rule</td>
<td>Select or clear the check box in the rule’s Enabled column.</td>
</tr>
<tr>
<td>Delete a Data Quality rule</td>
<td>Select the rule and then click the Delete button above the rules list. When prompted to confirm the deletion, click Yes.</td>
</tr>
<tr>
<td>Search for a Data Quality rule</td>
<td>Enter a search term in the Search box above the rules list.</td>
</tr>
<tr>
<td>Change the order of a rule</td>
<td>Select the rule that you want to move and then use the arrows above the rules list to change the position of the rule.</td>
</tr>
</tbody>
</table>

Table 9.4 | Managing Data Quality Rules
See also Error Mart.

Viewing Missing References

In some cases, incoming data is dependent on or refers to other data. If the referenced data cannot be loaded for some reason, you can either decide to add the data manually or continue on the assumption that the data will arrive before it is needed.

There are two way you can view missing references in Compose. Either via the Monitor tab in the Manage ETL Sets window or by switching the console to Monitor view and selecting the Missing References tab. The instructions below cover both of these methods.

To check for missing references in the Manage ETL Sets window

1. Click the Manage button in the lower left corner of the Data Warehouse panel.
2. Select the desired ETL set in the left side of the Manage ETL Sets window.
3. Switch to Monitor view by clicking the Monitor tab in the top right of the Manage ETL Sets window.
4. Click the View Missing References toolbar button.
   The Missing References - <ETL Set Name> window opens.
   The following information is displayed:
   **General information:** The run number of the ETL task, when it started and ended, the total number of inserts and updates, and the number of reported rows (if any).

   **Missing references information:**
   » **Missing Records from Entity** - The name of the entity with missing reference and the number of missing references.
      To see the missing record keys for the entity, click the number in parentheses to the right of the entity name.
      The Missing Record Keys for Entity - <Entity Name> window opens showing the list of missing keys and the number of times each key is referenced per entity.
   » **Referenced from Entity** - The entities that are referencing the entity with missing references.
   » **Via Relationship** - The name of the relationship in the Model.

5. To close the window, click Close.

To check for missing references in the Compose Monitor

1. Switch the console to Monitor View.
2. Select the desired data warehouse ETL task.
3. Click the Missing References tab below the task list.
   The following information is displayed:
   **General information:** The run number of the ETL task, when it started and ended, the total number of inserts and updates, and the number of reported rows (if any).
Missing references information:

» **Missing Records from Entity** - The name of the entity with missing reference and the number of missing references.

To see the missing record keys for the entity, click the number in parentheses to the right of the entity name.

The **Missing Record Keys for Entity** - `<Entity Name>` window opens showing the list of missing keys and the number of times each key is referenced per entity.

» **Referenced from Entity** - The entities that are referencing the entity with missing references.

» **Via Relationship** - The name of the relationship in the Model.

4. To close the window, click **Close**.

Missing References Example

In the following example, **Orders** and **Disputes** both reference **Customers**.

**Orders** contains seven records pointing to Mr. Brown and one record pointing to Mr. Smith. **Disputes** contains four records referencing Mr. Brown. Mr. Brown and Mr. Smith are "missing" from **Customers**.

The would be reflected as follows:

<table>
<thead>
<tr>
<th>Missing Records from Entity</th>
<th>Referenced from Entity</th>
<th>Via Relationship</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customers (2)</td>
<td>Orders (8)</td>
<td>Customers</td>
</tr>
<tr>
<td></td>
<td>Disputes (4)</td>
<td>CustomerDisputes</td>
</tr>
</tbody>
</table>

Clicking the number to the right of **Customers** (in the **Missing Records from Entity** column) would open the following window:

<table>
<thead>
<tr>
<th>Key</th>
<th>Referenced from Entity</th>
<th>Via Relationship</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mr. Brown</td>
<td>Orders (7)</td>
<td>Customers</td>
</tr>
<tr>
<td></td>
<td>Disputes (4)</td>
<td>CustomerDisputes</td>
</tr>
<tr>
<td>Mr. Smith</td>
<td>Orders (1)</td>
<td>Customers</td>
</tr>
</tbody>
</table>

See also: [How Compose Handles Missing References in the Data Warehouse](#).

Viewing and Exporting ETL Commands

You can view the ETL Commands that were run during the data warehouse ETL task. You can also export the ETL commands to a CSV file for reviewing and sharing.
To view the ETL Commands:
Click the **ETL Commands** toolbar button.

The **ETL - Name** dialog box opens in **List View**. Navigate through the commands using the scroll bar or find specific commands using the **Search** box.

-OR-

Click the **Item View** button and navigate through the commands using the navigation buttons at the bottom of the **ETL - Name** dialog box.

**Note**  
To jump to a specific command, type the command number in the **Go To** field at the bottom of the dialog box and then press [Enter].

To export the ETL Commands to a CSV file:
In **List View**, click the **Export to CSV File** button located to the left of the search field.

A file named "<name>_ETL_Instructions.csv" will be saved to your default Downloads location or you will be prompted to save it (according to your browser settings).

Modifying ETL Settings

For each ETL set, you can modify the settings according to your needs.

**To modify ETL settings:**
1. In the **Manage ETL Sets** window, select an ETL set in the left pane and then click **Settings**.
   
The **Setting - <ETL Set Name>** dialog box opens.
2. In the **General** tab, edit any of the following settings:
   
   » **Sequential Processing**: Select this option if you want all the ETL data warehouse processes to run sequentially, even if they can be run in parallel. This may be useful for debugging or profiling, but it may also affect performance.

   » **Default History Resolution**: Specifies at what resolution level the history timestamp is saved. If no mapping exists for the From Date (FD) column to the source, Attunity Compose uses **Minutes**. At this level, the timestamp includes the hours and minutes of when the change occurred. If you select a value of **Days**, the FD timestamp includes only the date, not the hours and minutes. If a mapping exists, then the FD level comes from the source.

   » In the **Advanced** tab, edit any of the following settings:

   » **Maximum number of database connections**: Enter the maximum number of connections allowed. The default size is 10.

   For more information, see Determining the Required Number of Database Connections.
» **Log level:** Select the log level granularity, which can be any of the following:

» **INFO (default)** - Logs informational messages that highlight the progress of the ETL process at a coarse-grained level.

» **TRACE** - Logs fine-grained informational events that can be used to debug the ETL process.

» **DEBUG** - Logs finer-grained informational events than the **TRACE** level.

Note that the log levels **DEBUG** and **TRACE** impact performance. You should only select them for troubleshooting if advised by Attunity Support.

» **JVM memory settings:** Edit the memory for the java virtual machine (JVM) if you experience performance issues. **Xms** is the minimum memory; **Xmx** is the maximum memory. The JVM starts running with the Xms value and can use up to the Xmx value.

» **Position in default workflow:** Select where you want the data warehouse tasks to appear in the default workflow. For more information on workflows, see **Workflows**.

» **Optimize for initial load:** Optimizes initial load in certain cases. Only select this option if the source tables do not reference missing records, use lookups, map different source records to the same record, or contain historical records. Note also that when this option is selected, data quality rules, derived attributes and the "Handle duplicates" option are not supported. In the event that the ETL set is used for incremental Full Load (using query-based change processing), clear the check box after the initial load task completes and regenerate the ETL set.

» **Write ETL command duration to the TLOG_PROCLOG table in the data warehouse:** This option is useful for troubleshooting performance issues with ETL processes as it records the duration of each ETL command in a special table (named TLOG_PROCLOG) in the data warehouse. You can then use this information to locate ETL commands with abnormal duration times and modify them accordingly.

» **Do not create indexes for data warehouse tables:** During an ETL Full Load task, Compose creates an internal index for each of the Data Warehouse tables (for query optimization). When running several consecutive ETL tasks (e.g. via a workflow) with a large volume of tables, this process can be extremely time-consuming. In such a scenario, best practice is to select the check box for each of the ETL tasks, except the last one. Note that this also requires the **Optimize for initial load** option (described above) to be selected.

» **Do not truncate staging tables:** Select this option if you want the ETL process to preserve the staging tables. Only use for debugging.

» **Stop processing after populating the staging tables:** Select if you do not want to proceed to populating the warehouse. Only use for debugging.

» **Do not drop temporary tables:** Select this option if you want to keep the temporary tables created during the ETL process. Only use for debugging.

When updating a non-null data warehouse column with a null value:
» **Do not change the target value:** Select this to keep values unchanged between two mappings for the same record. For instance, if the same record exists in two different source tables (A and B), but the record in Table A has a null value for data that is present in Table B (e.g. ZIP Code). In this case, if the record in Table A arrives after the record in Table B, the target value will be set to null. Selecting this option will prevent such an occurrence.

» **Set the target value to null:** Select this if you want the source and target values to correspond. This can be useful, for example, when a person moves address and one of the column values (e.g. "State") changes to null.

**Note** When ingesting changes from an Oracle source, this option requires full supplemental logging for all source table columns that exist on the target and any source columns referenced in filters, data quality rules, lookups, and expressions.

3. To save your changes, click **OK**.

**Validating the Data Warehouse**

Data warehouse validation should be performed each time the model is edited (after the data warehouse has already been created). Validating the data warehouse allows you automatically resolve any differences between the model and the data warehouse.

For a data warehouse to be considered valid, the tables defined in the data warehouse need to be identical to the physical tables in terms of metadata. Depending on the change, this may require adjusting the physical tables or dropping and recreating them (via Compose).

**Note** If the data warehouse is not valid, any ETL tasks that you attempt to run will fail.

Sometimes, however, the differences between the model and the data warehouse cannot be resolved automatically. In such cases, you need to drop and recreate the tables as described in **Dropping and Recreating Tables**.

**To validate the data warehouse:**

1. Click the **Validate** button at the bottom right of the Data Warehouse panel.
   - The **Validating the Data Warehouse** progress window opens.
   - If any differences are detected, the following message will be displayed:
     - The data warehouse is different from the model.
2. Click **Close**. The **Model and Data Warehouse Comparison Report** window opens.
3. Review the report and then click **Adjust Automatically** to resolve the differences automatically or **Generate Adjust Script** to generate a script with the adjust
commands.

**Note** The Adjust Automatically button will be disabled either if the Generate DDL scripts but do not run them option is selected or if Compose is unable to automatically adjust the data warehouse. In such cases, you should click Generate Adjust Script as described below.

Clicking Adjust Automatically:
If you clicked Adjust Automatically, the Adjust Data Warehouse progress window opens.
When the "The data warehouse was adjusted successfully." message is displayed, close the window. Note that adjusting the data warehouse may require you to update the data mart. In such a case, an appropriate message will be displayed for each of the data marts that require updating.

Clicking Generate Adjust Script:
Cases where Compose is unable to automatically adjust the data warehouse are as follows:
- A data type change that is not supported by the database or a data type change that may result in data loss.
- A change in an entity’s business key or distribution key.
- An attribute’s history type is Type 2 and the satellite table number in the attribute’s settings has changed.

When you click Generate Adjust Script, the Generate DDL Scripts window opens showing the progress of the script generation.
The generated scripts will be saved to:
`<product_dir>\data\projects\<project_name>\ddl-scripts`
Once the script(s) have been generated, close the Generate DDL Scripts window.
After you close the Generate DDL Scripts window, the DDL Script Files window opens automatically displaying the generated scripts. The DDL Script Files provides a read-only view that allows you to review the scripts and download them.
The scripts need to be executed directly in your data warehouse. Make sure that any modifications that you make to the scripts are done prior to executing them.

**Important:** When you run the adjust scripts, backup tables are created from the existing tables. The backup table names are appended with an ".old" suffix and must be deleted manually after the script completes.

**Note** Search for "TODO" in the script to locate the part of the script that needs modifying.
10 | Creating and Managing Data Marts

This section explains how to create data marts from your data warehouse tables.

In this chapter:
- Adding Data Marts and Star Schemas
- Displaying Data in a Pivot Table
- Managing Data Marts
- Pre and Post Loading ETLs
- Viewing and Exporting ETL Commands
- Validating the Data Mart
- Modifying Data Mart Settings

Adding Data Marts and Star Schemas

This topic explains how to create and manage data marts and star schemas in Attunity Compose. Since a data mart is essentially a subset of the data warehouse, you can create any number of data marts according to your BI needs. You can also create multiple star schemas for a single data mart. Star schemas allow you to reuse existing dimension tables within the same data mart, thereby saving space in the data warehouse while at the same time improving query performance. For example, you could create one star schema with an Order Details fact table and Customers and Products dimensions and another star schema with the same dimensions but a different fact type (or the same fact type, but different dimensions). This also allows you to generate BI reports using different facts that share the same dimensions. Additionally, in a star schema, dimensions are linked with each other through one join path intersecting the fact table, facilitating accurate and consistent query results.

A new data mart should be created in the following situations:
- Setting up a Compose project for the first time.
- To serve the needs of each individual business unit (different data marts can be used to obtain specific information for various enterprise departments, such as accounting, marketing, sales, and so on).

To create a data mart with a star schema:
1. Click the New button located at the bottom of the Data Mart panel.
   - OR -
Click the **Manage** button and then click the **New** button located at the top of the **Manage Data Marts** dialog box.

The **New Data Mart** dialog box opens.

2. Optionally change the default name and provide a description.

3. Make sure that the **Start New Star Schema Wizard** check box is selected (the default) and then click **OK**.

The **New Star Schema** wizard opens.

4. Provide a name and description (optional) for the star schema.

5. Select one of the available fact types:

   » **Transactional** - A star schema with a transactional fact table allows you to retrieve the desired data, even if a dimension table contains multiple versions of the same record. To use an example from the automotive industry, selecting "OrderDate" as the Transaction Date would allow you to generate a report for the number of customers who bought cars in New York between 2013 and 2016, even if a customer moved to a different city (which would also result in a new record being added to the Customers dimension).

   **Important:** When the Compose project is set up to use a Teradata data warehouse, an additional **Virtual Fact** check box is displayed. If you select this check box, the fact will be created as a view instead of a physical table. Virtual facts save space but can be slower to access than physical facts. You should therefore only select this option if you need to conserve storage space and do not need optimal access speeds.

   Additionally, if you select the **Virtual Fact** check box and the data mart and data warehouse are in different schemas, you need to grant the following permission:

   ```
   Grant SELECT on DWH_SCHEMA to DM_SCHEMA WITH GRANT OPTION
   ```

   » **Aggregated** - A star schema with an aggregated fact table allows you to make aggregate calculations based on the fact table attributes. For instance, you could create an aggregated fact that shows the total freight costs per shipping region and product category. Additionally, the presence of a transaction date in the fact table makes it possible to retrieve the desired data, even if a dimension contains multiple versions of the same record. To use an example from the shipping industry, a shipper could use an aggregated fact to generate a report for the total cost of shipping rice to Australia from 2015-2016.

   » **State Oriented** - A star schema with a state oriented fact supports Type 2 columns in the fact table. This is useful in cases where the fact is not a singular event in time, but rather, consists of multiple "states" or events that occur over time. Typical example of facts with multiple states are insurance claims or flight reservations. There are also cases when the same entity is treated as both a fact and a dimension - for example, Customers. In such cases, a report could be generated that relates to
the state of the fact, such as the time a claim was submitted to the time it was approved.

6. Click **Next**.

7. In the **Facts** screen, choose one fact for the star schema and then click **Next**.

   **Note**  If you do not want dimensions to be automatically selected in the **Dimensions** screen, clear the **Auto-select dimensions related to the selected fact(s)** check box at the bottom of the screen before clicking **Next**.

The **Dimensions** screen is displayed.

The left pane lists the dimensions that can be selected while the right pane displays a diagram of the star schema with the selected dimensions. You can view a dimension’s lineage by selecting the desired dimension and then clicking the **Lineage** button. For more information on lineage, see **Lineage and Impact Analysis**.

The left pane of the **Dimensions** screen contains the following areas:

- **Existing Dimensions** - Lists the dimensions that already exist in your data mart.
  
  Note that only dimensions that are relevant to the selected fact table will be displayed.

- **Create New Dimensions** - Lists all of the dimensions that can be added to the star schema.

- **Date Dimensions** - Lists all of the Date dimensions that can be added to the star schema. Note that these dimensions will only be available for selection if you added the Date and Time entities to your model.
  
  For an explanation of how to do this, see **Adding Date and Time Entities to your Model**.

- **Time Dimensions** - Lists all of the Time dimensions that can be added to the star schema. Note that these dimensions will only be available for selection if you added the Date and Time entities to your model.
  
  For an explanation of how to do this, see **Adding Date and Time Entities to your Model**.

   **Note**  When adding dimensions using the wizard, if a root dimension already exists in the data mart, any dimensions selected under the root dimension will be ignored.

   **Workaround:**

   Edit the dimension and delete or add columns as required.

8. Choose which dimensions to include in the star schema and then click **Next**.
9. If you chose **Star Schema with State Orientation** as your star schema type, click **Finish**. Otherwise, continue from Step Adding Data Marts and Star Schemas below.

10. In the **Transaction Date** screen, choose which Transaction Date to include in the data mart fact table. Selecting a Transaction Date enables you to retrieve the required data, even if the Dimension table contains multiple versions of the same record. For example, lets say that a car salesman wants to know how many customers bought cars in New York between 2013 and 2015. Selecting OrderDate as the Transaction Date for the Customers Dimension would make it possible to retrieve this information, even if a customer moved to a different city (which would also result in a new record being added to the data mart).

11. If you chose **Transactional** as your star schema fact type, click **Finish**. If you chose **Aggregated** as your star schema fact type, continue from Step Adding Data Marts and Star Schemas below.

12. In the **Aggregated Fact** screen:
   a. Select one or more columns from the Fact table on the left of the screen.

   **Note** You can select multiple columns by holding down the [Shift] (sequential selection) or [Ctrl] (non-sequential selection) buttons while selecting the columns.

   b. To add the column(s) to the **Group By** list on the right, either drag the columns to the list or click the arrowhead button to the left of the **Group By** list. Note that each dimension has a default "Group By" column that cannot be deleted.
   c. To add the column(s) to the **Aggregations** list on the right, either drag the columns to the list or click the arrowhead button to the left of the **Aggregations** list.
   d. To add new columns to the **Group By** or **Aggregations** list, click the **New** button above the list. In the **New column** dialog box, specify a **Name, Type, Description** and **Aggregation** (when adding a new aggregation column) and then click **OK**. The column is added to the list.
   e. To add an expression, hover the mouse cursor over the table cell in the **Expression** column and then click the function button that appears on the right. 

   **The Edit Expression: <Name>** dialog box opens.

   For more information on creating expressions, see Creating Expressions.
   f. To delete a column, select the column in the list and then click the **Delete** button above the list.
**Note** You can select multiple columns for deletion by holding down the [Shift] (sequential selection) or [Ctrl] (non-sequential selection) buttons while selecting the columns.

See also *Aggregation Example*.

13. Click **Finish**.

The newly created star schema is displayed below the **Star Schemas** heading, as shown below.

```
<table>
<thead>
<tr>
<th>Data Mart Model</th>
<th>Pre Loading</th>
<th>Post Loading</th>
</tr>
</thead>
<tbody>
<tr>
<td>☀ New Star Schema...</td>
<td>☐ New Dimension...</td>
<td>☐ Pivot</td>
</tr>
</tbody>
</table>

▲ Star Schemas
  - ☑ Star_Schema_1
  - ☑ Star_Schema_2

▲ Dimensions
  - ☑ Customers
  - ☑ Products
  - ☑ Employees
  - ☑ ShipVia_Shippers
  - ☑ Order Details
```

14. Click the **Create Tables** toolbar button.

The **Creating Data Mart: Data Mart Name in Target** progress window opens. Wait for the "Create Data Mart tables finished successfully." message to be displayed and then click **Close**.

**Note** After the data mart tables are created, the **Create Tables** button changes to **Drop and Recreate** tables.

15. Click the **Generate** toolbar button.

The **Generating Instructions for ETL Set: Data Mart Name** window opens. Wait for the "Generating ETL for Data Mart No. <number> finished successfully." message to be displayed and then click **Close**.

16. Click the **Run** toolbar button. The window switches to Monitor view and a progress bar
shows the current progress in terms of percentage.

When the Total ETL reaches 100 percent, data mart population is complete. You can stop the task at any time by clicking the Abort toolbar button. This may be necessary if you need to urgently edit the task settings due to some unforeseen development. After editing the task settings, simply click the Run button again to restart the task.

Note Aborting a task may leave the data warehouse tables in an inconsistent state. Consistency will be restored the next time the task is run.

Other monitoring information such as the run details (i.e. the number of rows inserted/updated) and the task log files can be accessed by clicking the Run Details and Log buttons respectively.

Should any errors occur, you can click the link at the end of the Failed bar for additional information that may help you troubleshoot the problem.

Once your data mart has been loaded with data, you can check that the required data is available for your BI tools. For more information, see Displaying Data in a Pivot Table.

Understanding Star Schema Icons
Compose displays various icons to indicate both the status and characteristics of the star schema tables. These icons are displayed in the table below.

<table>
<thead>
<tr>
<th>Icon</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>![Icon]</td>
<td>Indicates that although the structure for the star schema has been defined, all or part of the dimension(s) and/or fact table do not physically exist in the data warehouse. Click Create Tables to create</td>
</tr>
</tbody>
</table>
the tables and/or click **Validate** to see what needs to be adjusted.

| ![Checkmark] | Displayed when the dimension(s) and/or fact table physically exist in the data warehouse. |
| ![Star] | Indicates a conformed (shared) dimension. |
| ![Orders] | Small squares indicate that there are denormalized tables under the root dimension table. Each square represents a denormalized table, so in the image on the left, the **Orders** root dimension has four denormalized tables. To view the denormalized table names, hover the mouse cursor over each of the squares. |
| ![Refresh] | Displayed when a dimension has a reference to itself. |

### Aggregation Example

In the following example, Mike the organization’s data scientist, wants to create an aggregation table that shows the total freight costs per shipping region and product category; for example, the total cost of shipping rice to Australia in 2015.

To achieve this objective, he adds the **CategoryName** and **ShipRegion** attributes to the **Group By** list and then adds the **Freight** attribute to the **Aggregations** list. As Mike is interested in the total freight cost, he selects **SUM** as the **Aggregation Type**.
Displaying Data in a Pivot Table

This section explains how you can use Compose to view the data in your star schema.

**To view the data in a star schema:**

1. Click the **Manage** button at the bottom of the **Data Marts** panel.
2. In the **Manage Data Marts** dialog box, either:
   - Switch to Monitor view (by clicking the monitor icon) in the top right corner.
   - OR-
   - Remain in Design view and select a star schema.
3. Click the **Pivot** toolbar button.
   - If you clicked the **Pivot** toolbar button in Monitor view and your data mart contains several star schemas, you will be prompted to select a star schema.
   - The **Select columns for Pivot table** dialog box opens.
   - The drop-down list at the top of the dialog box contains the Fact table and the Dimensions tables that were used to create the star schema.
   - The Fact table name is prefixed with "Fct_" while dimension table names are prefixed with "Dim_".
4. Make sure that "Fct_<FactName>" is selected in the drop-down list and then select which fact column to add to the Pivot table.

5. From the drop-down list, select a dimension and then select which dimension columns to add to the Pivot table.

   **Note** If you added the Date and Time dimension tables to your data mart, you will be able to select.

6. Optionally, repeat Step 5 to add columns from different dimensions to the Pivot table.

7. Click **OK**. The pivot table dialog box opens.

   The names of columns that you can use to generate the data will be displayed at the top of the dialog box.

8. To form the actual table, drag columns to the gray area below the column names (the X-axis) and to the gray area on the left of the dialog box (the Y-axis).

   In figure **Pivot Table Example**, the **ShippedDate** column has been dragged to the X-axis while the **OrderID** column has been dragged to the Y-axis.

   In figure **Pivot Table with Quarter Example**, the QTR column was selected from the **Date** dimension, allowing orders to be grouped by quarter.
9. Change the table format, set aggregation, or perform additional actions as described in the table below.

<table>
<thead>
<tr>
<th>To</th>
<th>Do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set the</td>
<td>From the upper drop-down list in the left of the pivot table dialog box,</td>
</tr>
<tr>
<td>table</td>
<td>choose one of the following:</td>
</tr>
<tr>
<td>format</td>
<td>Table</td>
</tr>
</tbody>
</table>

Figure 10.2 | Pivot Table Example

Pivot Table with Quarter Example
To | Do this
--- | ---
| » Table bar chart  
| » Heatmap  
| » Row heatmap  
| » Col heatmap  
| » Treemap

**Aggregation options**
From the lower drop-down list in the left of the pivot table dialog box, choose one of the available options.
Note that additional drop-down lists may be displayed depending on the selected aggregation option. For example, when **Sum over Sum** is selected, two additional drop-down lists (containing column names) will appear below the aggregation options. The Sum over Sum aggregate is calculated by selecting one column from each of the drop-down lists.

**Change the columns**
Click the **Customize Columns** button and continue from Step 3 above.

10. Click **OK** to close the dialog box.

## Managing Data Marts
This topic describes the following management options:
» **Adding Star Schemas and Dimensions**
» **Editing Star Schemas**
» **Editing Dimensions**
» **Deleting Data Marts, Schemas and Dimensions**

### Adding Star Schemas and Dimensions
A data mart can contain any number of star schemas and dimensions. You can either add dimensions when you create a new star schema or you can add them later and attach them to star schemas as needed. Regardless of how they are added, dimensions can be reused across several star schemas as necessary.

**To add a star schema:**
1. Either click the **New Star Schema** toolbar button.
   - OR-
   Right-click the **Star Schemas** or **Dimensions** items and select **New Star Schema**.
The **New Star Schema** wizard opens.
2. Perform steps 4 to 13 in Adding Data Marts and Star Schemas. The star schema is added to the Star Schemas list.
3. If you already created the data mart tables (as described in Adding Data Marts and Star Schemas), you need to create the new star schema tables in the data mart. To do this, perform the validation process described in Validating the Data Mart. Otherwise, perform steps 4 to 13 in Adding Data Marts and Star Schemas. If you also want to run a data mart ETL task, perform step 16 as well.

To add a dimension:
1. Either click the New Dimension toolbar button.
   - OR-
   Right-click the Star Schemas or Dimensions items and select New Dimension. The Data Mart Name - New Dimension dialog box opens.
2. Select the dimension(s) you want to add to the data mart. Then click OK. The dimension(s) are added to the Dimensions list.
3. If you already created the data mart tables (as described in Adding Data Marts and Star Schemas), you need to create the new dimension table(s) in the data mart. To do this, perform the validation process described in Validating the Data Mart. Otherwise, perform steps 14 and 15 in Adding Data Marts and Star Schemas. If you also want to run a data mart ETL task, perform step 16 as well.

To attach a newly added dimension to a star schema:
1. Perform Steps 1-2 described in To add a dimension: above.
2. Select the dimension(s) you want to add to the star schema and then click the Add Dimension to Star Schema toolbar button. The Add Dimension <Name> to Start Schema dialog box opens.
3. Select which start schema(s) you want to add the dimension to and then click OK. The dimension is attached to the selected star schema(s).
4. If you already created the data mart tables (as described in Adding Data Marts and Star Schemas), you need to create the new dimension table(s) in the data mart. To do this, perform the validation process described in Validating the Data Mart. Otherwise, perform steps 14 and 15 in Adding Data Marts and Star Schemas. If you also want to run a data mart ETL task, perform step 16 as well.

Importing Dimensions
You can import dimensions from other data marts in the same project. This is especially useful in the following situations:

- Several developers are working on the same data mart, developing different complex dimensions
- When you need to use a dimension from another data mart and modify it slightly
To import dimensions

1. Click the Import Dimensions button in the lower toolbar
   The Import Dimensions dialog box opens.
2. From the Import from Data Mart drop-down list, select the data mart containing the
dimensions to import.
3. Select which dimensions to import and then click OK.

   Note: Only dimensions that do not already exist in the current data mart (with same
   name) are available for selection.

The dimensions are imported to your data mart.

Editing Star Schemas
You can edit a star schema according to your needs. Editing options include adding
columns, adding attributes and defining filters.

To edit a star schema (fact table):

1. Click the Manage button in the bottom left of the Data Mart panel.
   The Manage Data Marts dialog box opens.
2. In the left pane, select the data mart containing the star schema you want to edit.
3. Expand the list of start schemas and select the star schema you want to edit. Then either
click the Edit button in the lower toolbar or right-click the star schema and select Edit.
   The Edit Star Schema - Name dialog box opens. The following tabs are displayed:
   » General tab: In the General tab, you can edit the star schema name, the fact table
      name, the fact view name and the description.
      See also Data Mart Views.
   » Logical Attributes tab: In the Logical Attributes tab, you can add and delete
      columns, edit a column’s properties, view a column’s lineage, change the column
      order, and define filters.
      Edit the Logical Attributes tab according to Logical Attributes Tab Options.
   » Physical Table tab: The Physical Table tab provides a preview of the actual
     "physical" columns that will be created in the database.
     All editing tasks are performed in the Logical Attributes tab, except for the
     following tasks which are performed in the Physical Table tab:
     » Designate a Distribution Key Column (Amazon Redshift Only)
     » Designate Primary Index Column(s) (Teradata only)
     For more information, see Designating Distribution Key and Primary Index Columns.
   » Transaction Date tab: The Transaction Date tab enables you to change the
      transaction date that you selected when you created the star schema.
For more information on transaction dates, see the Transaction Date screen.

**Note** This tab will not be displayed if your Star Schema Type is "State Oriented".

### Table 10.1 | Logical Attributes Tab Options

<table>
<thead>
<tr>
<th>To</th>
<th>Do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delete a column</td>
<td>Select the column(s) you want to delete (multi-selection is supported) and click the Delete toolbar button.</td>
</tr>
<tr>
<td></td>
<td>The column(s) are deleted.</td>
</tr>
<tr>
<td>Add a new column</td>
<td>1. Click the New toolbar button.</td>
</tr>
<tr>
<td></td>
<td>The New Column dialog box opens.</td>
</tr>
<tr>
<td></td>
<td>2. In the Name field, specify a name for the column.</td>
</tr>
<tr>
<td></td>
<td>3. From the Type drop-down list, select one of the available data types.</td>
</tr>
<tr>
<td></td>
<td>4. If the selected data type requires further configuration, additional fields will be displayed. For example, when Decimal is selected, the Length and Scale fields will be displayed. Set the values as required.</td>
</tr>
<tr>
<td></td>
<td>5. Optionally specify a Description.</td>
</tr>
<tr>
<td></td>
<td>6. Click OK to add the column and close the New Column dialog box.</td>
</tr>
<tr>
<td>Edit a column’s properties</td>
<td>1. Double-click the row containing the column.</td>
</tr>
<tr>
<td></td>
<td>The Edit: Column Name dialog box opens.</td>
</tr>
<tr>
<td></td>
<td>2. Edit the properties as described in steps 2-6 of Add a new column above.</td>
</tr>
<tr>
<td>View a Column’s Lineage</td>
<td>1. Select the desired column.</td>
</tr>
<tr>
<td></td>
<td>2. Click the Lineage toolbar button. A windows displaying the column’s lineage is displayed. For more information about lineages, see Lineage and Impact Analysis.</td>
</tr>
<tr>
<td>Create a filter</td>
<td>Click the Filter toolbar button. The Expression Builder opens with the heading: Edit Filter - TableName. For information on creating filters, see Creating Expressions.</td>
</tr>
</tbody>
</table>

**Note** The assumption is that columns that are used in the filters do not change between different versions of the record. If this is not the case, the Full rebuild option should be selected in the Data Mart settings. This assumption is also true for relationships; for example, if a Sales record relates to Product which relates to Country, and the filter is applied to the product’s country, then the assumption is that the sale cannot change its product so that it is filtered in or out based on a new country.
Table 10.1 | Logical Attributes Tab Options

<table>
<thead>
<tr>
<th>To</th>
<th>Do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create or edit an expression</td>
<td>Hover the mouse cursor over the desired table column and then click the \textit{fx} button that appears to the right of the \textit{Expression} column. The Expression Builder opens with the heading: \textit{Edit Expression - Column Name}. For information on creating an expression, see \textit{Creating Expressions}.</td>
</tr>
<tr>
<td>Change the column order</td>
<td>Select the column(s) you want to move and then click the \textit{Move Down/Move to Bottom} or \textit{Move Up/Move to Top} buttons as desired.</td>
</tr>
</tbody>
</table>

Setting Sort Keys

You can define one or more of the physical table columns as sort keys. Amazon Redshift stores your data on disk in sorted order according to the sort key. The Amazon Redshift query optimizer uses sort order when it determines optimal query plans. For guidelines on choosing sort keys, visit http://docs.aws.amazon.com/redshift/latest/dg/c_best-practices-sort-key.html.

To add a sort key

1. Select the \textbf{Sort Keys} tab below the Columns list.
2. From the \textbf{Sort key style} drop-down list, choose one of the following styles:
   - \textbf{None} to disable the sort keys
   - \textbf{Compound} to use all of the columns listed in the sort key definition, in the order they are listed
   - \textbf{Interleaved} to give equal weight to each column in the sort key
3. Click the \textbf{Add Sort Key} button.
   A new row is added to the \textbf{Sort Keys} list. The \textbf{Position} column indicates the order of the column.
4. From the drop-down list in the \textbf{Column} column, select the desired column.
   The column is add to the list.
5. Click \textbf{OK} to save your settings and close the \textbf{Edit Dimension/Edit Star Schema} window.

Managing Sort Keys

You can delete sort keys and change their position as described in the table below.
To Do this
Delete a sort key Select the sort key you want to delete and then click the Delete button.
Change the position of a sort key Select the sort key you want to move and then click the up or down arrows to promote or demote the key.

For more information about sort keys, visit:
http://docs.aws.amazon.com/redshift/latest/dg/t_Sorting_data.html

Editing Dimensions
You can edit a dimension according to your needs. Editing options include adding columns, adding attributes and defining filters.

To edit a dimension:
1. Click the Manage button in the bottom left of the Data Mart panel.
   The Manage Data Marts dialog box opens.
2. In the left pane, select the data mart containing the star schema you want to edit.
3. Expand the list of dimensions and select the dimension you want to edit. Then either click the Edit button in the lower toolbar or right-click the star schema and select Edit.
   The Edit Conformed Dimension - Name (or Edit Dimension - Name if the dimension has not yet been added to the data mart) dialog box opens. The following tabs are displayed:
   - General tab: In the General tab, you can edit the dimension name, the dimension table name, the dimension view name and the description. You can also change the dimension’s history type by selecting Type 1 or Type 2 from the History Type drop-down list. For more information on changing the history type, see Understanding Dimension History Types.
     See also Data Mart Views.
   - Logical Attributes tab: In the Logical Attributes tab, you can add and delete columns, edit a column’s properties, view a column’s lineage, change the column order, and define filters.
     Edit the Logical Attributes tab according to Logical Attributes Tab Options.
   - Physical Table tab: The Physical Table tab provides a preview of the actual "physical" columns that will be created in the database.
     All editing tasks are performed in the Logical Attributes tab, except for the following tasks which are performed in the Physical Table tab:
     - Designate a Distribution Key Column (Amazon Redshift only)
     - Setting Sort Keys (Amazon Redshift only)
     - Designate Primary Index Column(s) (Teradata only)
For information on designating distribution key and primary index columns, see Designating Distribution Key and Primary Index Columns.

4. Edit the Logical Attributes tab according to Logical Attributes Tab Options.

**Note** You can apply or revert your changes at any time, simply by clicking the Apply or Cancel buttons respectively.

5. Click OK to close the dialog box and save your settings or Cancel to close the dialog box without saving your settings.

### Setting Sort Keys

You can define one or more of the physical table columns as sort keys. Amazon Redshift stores your data on disk in sorted order according to the sort key. The Amazon Redshift query optimizer uses sort order when it determines optimal query plans. For guidelines on choosing sort keys, visit http://docs.aws.amazon.com/redshift/latest/dg/c_best-practices-sort-key.html.

#### To add a sort key

1. Select the Sort Keys tab below the Columns list.
2. From the Sort key style drop-down list, choose one of the following styles:
   - **None** to disable the sort keys
   - **Compound** to use all of the columns listed in the sort key definition, in the order they are listed
   - **Interleaved** to give equal weight to each column in the sort key
3. Click the Add Sort Key button.
   - A new row is added to the Sort Keys list. The Position column indicates the order of the column.
4. From the drop-down list in the Column column, select the desired column.
   - The column is add to the list.
5. Click **OK** to save your settings and close the Edit Dimension/Edit Star Schema window.

### Managing Sort Keys

You can delete sort keys and change their position as described in the table below.

<table>
<thead>
<tr>
<th>To</th>
<th>Do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delete a sort key</td>
<td>Select the sort key you want to delete and then click the Delete button.</td>
</tr>
<tr>
<td>Change the position of a sort key</td>
<td>Select the sort key you want to move and then click the up or down arrows to promote or demote the key.</td>
</tr>
</tbody>
</table>
For more information about sort keys, visit:
http://docs.aws.amazon.com/redshift/latest/dg/t_Sorting_data.html

Understanding Dimension History Types
By default, dimension tables are defined as history type 2, meaning that a new record is added each time a record is updated (as opposed to updating the same record). In the data warehouse, dimension tables always contain an object identifier column which is the table name appended with "_OID". However, when a dimension table’s history type is 2, an additional "version" column is added as the dimension table’s Primary Key. This column, which is used to identify the version, has the same name as the table, but is appended with "VID".

The flowing figure shows the Customer_VID and Customer_OID (object identifier) columns in the Customers dimension table:

![Table Screenshot](image)

**Note** If a dimension table is defined as history type 1 and one or more of the columns in the corresponding data warehouse are defined as type 2, records in the dimension table will simply be replaced with the most up-to-date record in the corresponding data warehouse table.

Bulk Operations
You can perform the following bulk operations on one or more data marts:

- Drop and recreate tables
- Enable/Disable the Optimize for initial load option
- Generate the data mart ETL set

**To perform bulk operations**
1. Click the **Bulk Operations** toolbar button in the **Manage Data Marts** widow.
   The **Bulk Operations** dialog box opens.
2. Select which operations to perform and on which data marts to perform them.
3. Click OK.
   The Preparing All Data Marts window opens, displaying the progress of the selected operations.
4. When the <n> data marts were prepared successfully message is displayed, click Close.

Data Mart Views
By default, data mart tables are created without corresponding views. Views may be useful however because they:
» Allow authorized employees to utilize the data mart for analytics while preventing changes to the actual data.
» Can be queried without needing to worry about "locking" the data.
When you create a view for a fact or dimension table, you also need to set the schema in which the view will be created.
Additionally, the schema permissions should only allow authorized employees to access the view(s).
For instructions on setting the view schema, see Modifying Data Mart Settings.
See also Editing Star Schemas and Editing Dimensions.

Deleting Data Marts, Schemas and Dimensions
You can delete data marts, star schemas and dimensions that you no longer require.

<table>
<thead>
<tr>
<th>To</th>
<th>Do This</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delete a data mart</td>
<td>1. Select the data mart and then click the upper Delete Data Mart toolbar button.</td>
</tr>
<tr>
<td></td>
<td>2. Click OK when prompted to confirm the deletion.</td>
</tr>
<tr>
<td>To</td>
<td>Do This</td>
</tr>
<tr>
<td>--------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Delete a star schema</strong></td>
<td>1. Select the star schema and then click the lower <strong>Delete</strong> toolbar button. -OR- Right-click the star schema and select <strong>Delete</strong>.</td>
</tr>
<tr>
<td></td>
<td>2. Optionally select the <strong>Drop unused dimensions</strong> check box.</td>
</tr>
<tr>
<td></td>
<td>3. Click <strong>OK</strong> when prompted to confirm the deletion.</td>
</tr>
<tr>
<td><strong>Delete a dimension</strong></td>
<td>1. Select the dimension and then click the lower <strong>Delete</strong> toolbar button. -OR- Right-click the dimension and select <strong>Delete</strong>.</td>
</tr>
<tr>
<td></td>
<td>2. Click <strong>OK</strong> when prompted to confirm the deletion.</td>
</tr>
</tbody>
</table>

### Pre and Post Loading ETLs

You can create and manage pre and post loading ETLs as described in the table below.

<table>
<thead>
<tr>
<th>To</th>
<th>Do This</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Define a Pre Loading or Post Loading ETL</strong></td>
<td>1. Select either the <strong>Pre Loading</strong> or <strong>Post Loading</strong> tab as appropriate.</td>
</tr>
<tr>
<td></td>
<td>2. Click the <strong>New</strong> button above the User Defined ETL column.</td>
</tr>
<tr>
<td></td>
<td>The <strong>Add New Pre/Post Loading</strong> dialog box opens.</td>
</tr>
<tr>
<td></td>
<td>3. Specify a name for the ETL and then click <strong>OK</strong>.</td>
</tr>
<tr>
<td></td>
<td>The name is added as a link to the <strong>User-Defined ETL</strong> column.</td>
</tr>
<tr>
<td></td>
<td>4. Click the link.</td>
</tr>
<tr>
<td></td>
<td>The <strong>Edit ETL Instruction: Name</strong> dialog box opens.</td>
</tr>
<tr>
<td></td>
<td>5. To define an ETL:</td>
</tr>
<tr>
<td></td>
<td>a. Select a table and/or column and click the arrow to the right of the selected table/column to add it to the ETL.</td>
</tr>
<tr>
<td></td>
<td>b. Use the <strong>Select, Delete, Insert</strong> and <strong>Update</strong> quick access buttons at the top of the dialog box to add SQL statements to your ETL.</td>
</tr>
<tr>
<td></td>
<td>c. To save your ETL, click <strong>OK</strong>.</td>
</tr>
<tr>
<td></td>
<td>d. To enable/disable the ETL, select or clear the check box to the left of it as required.</td>
</tr>
</tbody>
</table>
## To Rename a Pre Loading or Post Loading ETL

<table>
<thead>
<tr>
<th>To</th>
<th>Do This</th>
</tr>
</thead>
</table>
| Rename a Pre Loading or Post Loading ETL | 1. Select either the **Pre Loading** or **Post Loading** tab as appropriate.  
2. At the end of the row containing the ETL name click the **Rename** button (A).  
3. Rename the ETL and then click **OK** to save the new name. |

## To Edit a Pre Loading or Post Loading ETL

<table>
<thead>
<tr>
<th>To</th>
<th>Do This</th>
</tr>
</thead>
</table>
| Edit a Pre Loading or Post Loading ETL | 1. Select either the **Pre Loading** or **Post Loading** tab as appropriate.  
2. At the end of the row containing the ETL name click the **Edit** button.  
3. Edit the ETL as described in Step 5 of **Define a Pre Loading or Post Loading ETL**. |

## Viewing and Exporting ETL Commands

You can view the ETL Commands that are run during the data mart ETL task. You can also export the ETL commands to a CSV file for reviewing and sharing.

**To view the ETL Commands:**

Click the **ETL Commands** toolbar button.

The **ETL - Name** dialog box opens in **List View**. Navigate through the commands using the scroll bar or find specific commands using the **Search** box.

-OR-

Click the **Item View** button and navigate through the commands using the navigation buttons at the bottom of the **ETL - Name** dialog box.

**Note** To jump to a specific command, type the command number in the **Go To** field at the bottom of the dialog box and then press [Enter].

**To export the ETL Commands to a CSV file:**

In **List View**, click the **Export to CSV File** button located to the left of the search field.

A file named "<name>_ETL_Instructions.csv" will be saved to your default Downloads location or you will be prompted to save it (according to your browser settings).

## Validating the Data Mart

Whenever you edit a data mart, certain actions need to be performed to ensure that the data mart is valid. For a data mart to be considered valid, the tables defined in the data
The data mart need to be identical to the physical tables in terms of metadata. Depending on the change, this may require adjusting the physical tables or dropping and recreating them (via Compose).

Additionally, the generated ETL commands must reflect the current state of the data mart. So, for example, if a filter or expression was added/edited, you will need to regenerate the ETL commands before running the data mart ETL task.

**Note** If the data mart is not valid, any ETL tasks that you attempt to run will fail.

Situations in which you need to validate the data mart and/or regenerate the ETL commands include:

- Each time the data warehouse is adjusted
- Each time a new dimension is added to a star schema
- Each time a new star schema is added to a data mart
- Adding or removing columns
- Changes to a dimension’s history type
- Changes to transformations (expressions/filters)
- Changes to a star schema’s transaction date
- Changes to a star schema’s aggregation type (max, min, etc.)

Note that clicking the **Validate** button only verifies that the table metadata is valid. In certain cases, even if the metadata is valid, Compose will prompt you to regenerate the ETL commands (by clicking the **Generate** button).

When you validate a data mart, Compose presents you with a list of operations that it needs to perform for the data mart to be valid. Examples of such operations include adding dimension and fact tables, deleting the fact table when the transaction date column has been deleted from the model, and so on. You can either click **Adjust Automatically** or **Drop and Recreate Tables** to approve the operations or click **Cancel** to continue working with the data mart in its present state.

**To validate (and adjust) the data mart:**

1. Click the **Validate** toolbar button in the **Manage Data Marts** dialog box.
   - The **Validating the Data Mart** progress window opens.
   - If any differences are detected, the following message will be displayed:
     
     Data mart validation failed. The data mart is different from the model.

2. Click **Close**.
   - The **Model and Data Mart Comparison Report** window opens.

3. Review the report and then click **Adjust Automatically** or **Drop and Recreate Tables** to resolve the differences.
Either the Adjust Data Mart progress window opens or, if you clicked Drop and Recreate Tables, confirm the drop and recreate operation. When you confirm the drop and recreate operation, the Creating Data Mart: Name window is displayed.

4. When the "The data mart was adjusted successfully." or "The data mart has been created successfully." (in the case of drop and recreate) message is displayed, close the window.

5. Click the Generate toolbar button to regenerate the ETL commands.

**Note** When a dimension’s history type has been changed directly in the data mart, the data mart validation will be successful, but you need to drop and recreate the tables by clicking the Create Tables toolbar button.

For information on changing history types, see the General tab tab in the Edit Dimensions dialog box.

### Modifying Data Mart Settings

You can modify the data mart settings according to your needs.

**To modify data mart settings:**

1. In the Manage Data Marts window, select a data mart and click Settings.

   The Setting - Data Mart Name dialog box opens.

   a. In the General tab, edit any of the following settings:

   » **Sequential Processing:** Select this option if you want to run all ETLs of the data warehouse sequentially, even if they can be run in parallel. This can be good for debugging or profiling.

   » **Load Type:** Select Full rebuild to build the data mart from scratch or Incremental loading (default) to only load changes. Note that Incremental loading is not currently available for aggregated fact tables.

   » **Create tables in database** - By default, data mart tables are created in the database specified in the data warehouse connection settings. Optionally, click the browse button and select a different database.

      **Note** This option is only available for Microsoft SQL Server.

   » **Create tables in schema** - By default, data mart tables are created in the schema specified in the data warehouse connection settings. Optionally, specify a different schema, either by typing the schema name or by clicking the browse button and selecting one of the existing schemas. If you specify the name of a non-existing schema, Compose will create the schema automatically.
Note  This option is only available for Microsoft SQL Server.

» Create views in schema - By default, data mart views are created in the schema specified in the data warehouse connection settings. Optionally, specify a different schema, either by typing the schema name or by clicking the browse button and selecting one of the existing schemas. If you specify the name of a non-existing schema, Compose will create the schema automatically, unless the database is Oracle or Teradata.

Note  If the view schema is different from the data mart schema, you need to grant the following permission:
Grant SELECT on DM_SCHEMA to DM_VIEW_SCHEMA WITH GRANT OPTION

See also Data Mart Views.

2. In the Advanced tab, edit any of the following settings:

» Maximum number of database connections: Enter the maximum number of connections allowed. The default size is 10.
For more information, see Determining the Required Number of Database Connections.

» Log level: Select the log level granularity, which can be any of the following:
  » INFO (default) - Logs informational messages that highlight the progress of the ETL process at a coarse-grained level.
  » TRACE - Logs fine-grained informational events that are most useful to debug the ETL process.
  » DEBUG - Logs finer-grained informational events than the TRACE level.
The log levels DEBUG and TRACE impact performance. You should only select them for troubleshooting if advised by Attunity Support.

» JVM memory settings: Edit the memory for the java virtual machine (JVM) if you experience performance issues. Xms is the minimum memory; Xmx is the maximum memory. The JVM starts running with the Xms value and can use up to the Xmx value.

» Position in Default Workflow: Set the position you want the data mart to appear in the default workflow. For more information on workflows, see Workflows.

» Do not drop temporary tables: Select this option if you want to keep the temporary tables created during the ETL process. Only use for debugging.

» Enable table logging: This option is available for Oracle only. When enabled, the data mart tables will be created with the Oracle LOGGING option enabled. Leaving this option unselected (the default) should improve performance, but in some cases DML operations will not be recorded in the redo log file. For more information on this option, refer to the Oracle online documentation.
» **Write ETL command duration to the TLOG_PROCLOG table in the data warehouse:** This option is useful for troubleshooting performance issues with ETL processes as it records the duration of each ETL command in a special table (named TLOG_PROCLOG) in the data warehouse. You can then use this information to locate ETL commands with abnormal duration times and modify them accordingly.

» **Optimizes initial load in certain cases:** This option is not applicable to the State Oriented and Aggregated fact types. If the "Optimize for initial load" option is enabled (the default), clear the "Incremental Loading" option after the initial load task completes and regenerate the Data Mart ETL set. If the "Full Rebuild" option is enabled, selecting "Optimize for initial load" may accelerate the loading process.

3. Click **OK**.
11 | Controlling and Monitoring Tasks and Workflows

The Compose monitor shows the current status of all your ETL tasks and enables you to drill-down for additional information about each task. Task instances can be run immediately or scheduled to run in the future (either once or at set intervals).

In this chapter:
- Viewing Information in the Monitor
- Viewing Missing References
- Controlling Tasks
- Defining Notifications Rules
- Workflows
- Monitoring and Controlling Replicate Tasks
Viewing Information in the Monitor

As well as providing a high-level summary of all your ETL tasks, the monitor also lets you view more detailed information about specific tasks.

**To switch to monitor view:**
1. Open a Compose project and click the **Monitor** icon in the top right of the console.

   A list of ETL tasks is displayed for the current project. The left pane of the monitor allows you to filter the task list by status as well as indicating the current number of running, failed and completed tasks.

For each task, the monitor displays the following information:

- **Status** - Running, Completed or Failed
- **Name** - The task name.
- **Type** - Data Warehouse, Data Mart, Workflow, Command Task or Replicate Task.
- **Started and Ended** - The date and time the task started and completed (according to the server time). If the task is running, the **Ended** column will display the current progress. In the case of a Replicate task performing Change Processing, **Running CDC** will be displayed.
- **Next Instance** - The next time the task is due to run (if the task is scheduled).
- **Elapsed Time** - The time it took for the task to complete or - if the task is still running - how long the task has been running.
- **Inserted Rows** - The number of rows inserted into the data warehouse or data mart.
- **Updated Rows** - The number of rows updated in the data warehouse or data mart.
- **Reported Rows** - The number of rows reported to the error mart table(s).
- For more information on error marts, see [Defining and Managing Data Quality Rules](#).
- **Disabled** - Whether the scheduled job has been disabled.
2. To view additional information about a task, select the task. The information is displayed in the following tabs in the lower pane of the monitor:

   - **Details** - Use the navigation arrows to browse through task instances. For each entity, the total number of inserted, updated and deleted rows is shown.
   - **History** - The History tab provides a list of previous task instances. To view a task instance’s log file, select the task and click the View Log button. To view more details about a task instance, either double-click the instance or select the instance and then click the View Instance Details button. The Details tab is shown.
   - **Progress Status** - The Progress Status tab shows the task’s current progress as well as the sub-status of ETL commands within the task (Waiting/Standby, Running, Failed, etc.). To see details about a specific ETL commands, click the number to the right of the command status. For example, to view more information about an ETL command with an error status, click the number to the right of Error.
   - **Error Mart** - The Error Mart tab displays information when one or more Data Quality rules are enforced during the ETL task (and the rules have been configured to report non-compliant data).
     The following information is available:
     - **Entity Name** - The name of the entity for which rule was created.
     - **Mappings** - The mapping for which the rule was created.
     - **Error Mart Table** - The name of the error mart table created as a result of the rule being enforced.
     - **Schema** - The name of the schema containing the error mart table.
     - **Reported Rows** - The number of rows reported to the error mart. Clicking the number opens the actual error mart table as it appears in the data warehouse.
   - **Missing References** - For a description of this tab, see Viewing Missing References. See also: Defining and Managing Data Quality Rules.

3. To run a job immediately, select the task and then click the Run toolbar button.
4. To view a task’s settings, select the task and then click the Open toolbar button. For more information about the settings, see Creating and Managing the Data Warehouse and Creating and Managing Data Marts.

**Viewing Missing References**

In some cases, incoming data is dependent on or refers to other data. If the referenced data cannot be loaded for some reason, you can either decide to add the data manually or continue on the assumption that the data will arrive before it is needed.

There are two way you can view missing references in Compose. Either via the Monitor tab in the Manage ETL Sets window or by switching the console to Monitor view and
selecting the **Missing References** tab. The instructions below cover both of these methods.

**To check for missing references in the Manage ETL Sets window**

1. Click the **Manage** button in the lower left corner of the **Data Warehouse** panel.
2. Select the desired ETL set in the left side of the **Manage ETL Sets** window.
3. Switch to Monitor view by clicking the **Monitor** tab in the top right of the **Manage ETL Sets** window.
4. Click the **View Missing References** toolbar button.
   
   The **Missing References - <ETL Set Name>** window opens.

   The following information is displayed:

   **General information:** The run number of the ETL task, when it started and ended, the total number of inserts and updates, and the number of reported rows (if any).

   **Missing references information:**

   » **Missing Records from Entity** - The name of the entity with missing reference and the number of missing references.

   To see the missing record keys for the entity, click the number in parentheses to the right of the entity name.

   The **Missing Record Keys for Entity - <Entity Name>** window opens showing the list of missing keys and the number of times each key is referenced per entity.

   » **Referenced from Entity** - The entities that are referencing the entity with missing references.

   » **Via Relationship** - The name of the relationship in the Model.

5. To close the window, click **Close**.

**To check for missing references in the Compose Monitor**

1. Switch the console to Monitor View.
2. Select the desired data warehouse ETL task.
3. Click the **Missing References** tab below the task list.

   The following information is displayed:

   **General information:** The run number of the ETL task, when it started and ended, the total number of inserts and updates, and the number of reported rows (if any).

   **Missing references information:**

   » **Missing Records from Entity** - The name of the entity with missing reference and the number of missing references.

   To see the missing record keys for the entity, click the number in parentheses to the right of the entity name.

   The **Missing Record Keys for Entity - <Entity Name>** window opens showing the list of missing keys and the number of times each key is referenced per entity.
- **Referenced from Entity** - The entities that are referencing the entity with missing references.
- **Via Relationship** - The name of the relationship in the Model.

4. To close the window, click **Close**.

### Missing References Example

In the following example, **Orders** and **Disputes** both reference **Customers**.

**Orders** contains seven records pointing to Mr. Brown and one record pointing to Mr. Smith. **Disputes** contains four records referencing Mr. Brown. Mr. Brown and Mr. Smith are "missing" from **Customers**.

The would be reflected as follows:

<table>
<thead>
<tr>
<th>Missing Records from Entity</th>
<th>Referenced from Entity</th>
<th>Via Relationship</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customers (2)</td>
<td>Orders (8)</td>
<td>Customers</td>
</tr>
<tr>
<td></td>
<td>Disputes (4)</td>
<td>CustomerDisputes</td>
</tr>
</tbody>
</table>

Clicking the number to the right of **Customers** (in the **Missing Records from Entity** column) would open the following window:

<table>
<thead>
<tr>
<th>Key</th>
<th>Referenced from Entity</th>
<th>Via Relationship</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mr. Brown</td>
<td>Orders (7)</td>
<td>Customers</td>
</tr>
<tr>
<td></td>
<td>Disputes (4)</td>
<td>CustomerDisputes</td>
</tr>
<tr>
<td>Mr. Smith</td>
<td>Orders (1)</td>
<td>Customers</td>
</tr>
</tbody>
</table>

See also: [How Compose Handles Missing References in the Data Warehouse](#).

### Controlling Tasks

You can run and stop tasks/workflow manually or you can schedule them using the scheduling options described in **Scheduling Tasks**.

**Running and Aborting Tasks Manually**

You can run tasks manually and abort them if required.

**To run a task manually:**

- Select the task and click the **Run** toolbar button.
To abort a task:

- Select the task and click the **Abort** toolbar button.

  The task process is aborted. Note that aborting a task may leave the data warehouse or data mart tables in an inconsistent state. Consistency will be restored the next time the task is run.

Scheduling Tasks

Scheduling tasks is a convenient way of continually updating the data warehouse and associated data mart(s). For instance, you could schedule a data warehouse ETL task to run at 4:00 pm and then schedule a data mart ETL task to run at 5:00 pm.

Note that as Compose currently does not provide a task-chaining option (i.e. run another task as soon as the current task completes), it may be better to schedule tasks using an external tool that supports this capability.

You can also use the command line interface (CLI) to run an ETL task. For details, see Running ETL Tasks using the CLI.

To schedule a task:

1. Click the **Schedule** toolbar button.
2. In the **<Name> Scheduler** dialog box, choose one of the following options from the **Run Job** drop-down list.
   - **Once** - to run the job once on a specific date and time.
   - **Every** - to run the job at set intervals.
   - **Daily** - to run the job every day at a specific time.
   - **Weekly** - to run the job on selected days at a specific time
   - **Monthly** - to run the job on the nth of every month at a specific time
   - **Advanced** - to use a Cron expression. For a description of allowed cron formats together with usage examples, see [Cron Format and Examples](https://crontab.guru/).
3. Set the scheduling parameters according to the selected scheduling option.
4. Click **OK** to save your settings.
   - The date and time the next instance is scheduled to run will appear in the **Next Instance** column.
5. To disable a scheduled job, select the task and click the **Edit Scheduling** toolbar button. Then, select the **Disable** check box in the **<Name> Scheduler** dialog box.
6. To cancel a scheduled job for a task, select the task and click the **Edit Scheduling** toolbar button. Then, in the **<Name> Scheduler** dialog box, click **Delete**.

Running ETL Tasks using the CLI

Before you can run a task, you must first run the **Connect** command as described in Connecting to Attunity Compose Server.
Note As Compose CLI requires Administrator permission, make sure to select "Run as administrator" when opening the command prompt.

The RunTask command populates the data warehouse or data mart with data. The "ETL" operation can also be performed using the Run toolbar button located in Monitor view as well as in the Manage ETL Set and Manage Data Marts dialog boxes.

When this command succeeds, it returns 0.

Syntax:
ComposeCli.exe run_task --project [project-name] --type [DW|DM] --task [task-name] --wait [timeout-in-sec]

where:
- **project** is the name of your Compose project
- **type** is the type of tables you want to populate. Specify **DW** to populate a data warehouse or **DM** to populate a data mart.
- **task** is the name of the ETL Set you want to run (when --type DW) or the name of the data mart you want to populate (when --type DM).
- **wait** is the wait time specified in seconds.

The command line can run in sync or async mode. A value of 0 (seconds) indicates sync mode. This means that as soon as the task finishes, the command line returns to prompt. The default mode is async, with a value of -1. This is also applied if you leave this parameter empty. Other negative values are not permitted.

Note that if **wait** is excluded from the command, the task may appear to complete successfully even if it encountered an error.

Example:
ComposeCli.exe run_task --project nw --task DWH_1 --type DW
Compose Control Program started...
Instance Number: 9
Compose Control Program completed successfully.

Defining Notifications Rules
You can select events, on the occurrence of which, a notification will be sent to the specified recipients.
To set a notification rule:

1. Switch to Monitor view.
2. Click the downward arrow at the top left of the console and select Notification Rules from the drop-down menu.
   The Notification Rules dialog box opens.
3. Click the New toolbar button.
   The New Notification wizard opens.
4. In the Events screen:
   » Specify a name for the notification
   » Choose for which type of events you want the notification to be sent, both at the task level and at the workflow level.
5. Click Next. In the Recipients screen:
   » Select Windows Event to send the notification to Windows Event Viewer and/or Recipients to send the notification to a list of email recipients.
   » If you selected Recipients, enter the recipient email address(es) in the To, Cc (optional) and Bcc (optional) boxes. Multiple addresses must be separated by a semi-colon.
6. Click Next. In the Message screen, optionally, edit the default notification message. You can add variables to the message by selecting the variable on the right and then clicking the arrow to the left of the variables list.
   The following variables are available:

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>${PROJECT}</td>
<td>The name of the Compose project in which the event occurred.</td>
</tr>
<tr>
<td>${TASK_NAME}</td>
<td>The name of the ETL task in which the event occurred.</td>
</tr>
<tr>
<td>${INSERTED}</td>
<td>The number of rows inserted in the Data Warehouse/Data Mart.</td>
</tr>
<tr>
<td>${UPDATED}</td>
<td>The number of rows updated in the Data Warehouse/Data Mart.</td>
</tr>
<tr>
<td>${DELETED}</td>
<td>The number of rows deleted from the Data Warehouse/Data Mart.</td>
</tr>
<tr>
<td>${ERROR_CODE}</td>
<td>The error code if an error was encountered during the task.</td>
</tr>
<tr>
<td>${ERROR_DETAILS}</td>
<td>The error message if an error was encountered during the task.</td>
</tr>
<tr>
<td>${EVENT_TYPE}</td>
<td>The event type (Started, Error or Completed).</td>
</tr>
</tbody>
</table>
### Variable Description

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>${EVENT_TYPE_DESCRIPTION}</td>
<td>The date and time the event occurred.</td>
</tr>
<tr>
<td>${EVENT_TIME}</td>
<td>A link to the relevant Compose project.</td>
</tr>
</tbody>
</table>

7. Click **Next**. In the **Associate with** screen, select whether to apply the rule to all tasks of to selected tasks. If you chose **Selected Tasks**, select which tasks to apply the rule to.

8. Click **Next** to see a summary of the notification settings or **Finish** to save your settings and exit the wizard.

9. If you clicked **Next**, review your settings and then click **Finish** to save the notification rule and exit the wizard or **Prev** to edit your settings. You can also click the headings on the right of the wizard to go directly to a specific dialog box.

   The notification will be added to the list of notifications in the Notification Rules dialog box.

### Managing Notification Rules

In the **Notification Rules** dialog box, you can edit, delete and enable/disable notification rules as described in the table below.

<table>
<thead>
<tr>
<th>To Do This</th>
<th>To Do This</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delete a Rule</td>
<td>Select the rule and then click the <strong>Delete</strong> toolbar button. When prompted to confirm the deletion, click <strong>Yes</strong>.</td>
</tr>
<tr>
<td>Edit a Rule</td>
<td>Either double-click the rule you want to edit or select the rule and click the <strong>Edit</strong> toolbar button. Continue from <strong>Defining Notifications Rules</strong>.</td>
</tr>
<tr>
<td>Disable a Rule</td>
<td>Select the rule you want to disable and then either click the <strong>Disable</strong> toolbar button or clear the check box in the <strong>Enabled</strong> column.</td>
</tr>
<tr>
<td>Enable a Rule</td>
<td>Select the rule you want to enable and then click the <strong>Enable</strong> toolbar button or select the check box in the <strong>Enabled</strong> column.</td>
</tr>
</tbody>
</table>

### Workflows

Workflows enable you to run tasks both sequentially and in parallel. You can either schedule workflows as described in **Scheduling Tasks** or run them manually using the **Run** toolbar button.
You can create your own workflow and/or use the built-in workflow. The built-in workflow enables you to run all of your data warehouse and data mart ETL tasks as a single, end-to-end process. The built-in workflow appears in the **Type** column as "Default Workflow". In a project with a single data mart, all tasks run sequentially in the default workflow, starting with the data warehouse ETL tasks and ending with the data mart ETL task. However, in projects with multiple data marts, the data mart ETL tasks run in parallel, following the completion of the data warehouse ETL tasks.

When you create your own workflow, you decide which tasks to include in the workflow and the order in which they will be run.

Creating Workflows
This section provides instructions for creating workflows.

**To create a workflow:**

1. Switch to Monitor view by clicking the **Monitor** button in the top right of the Compose console.
2. Click the **New Workflow** toolbar button. The **New Workflow** dialog box opens.
3. Specify a name for your workflow.
4. Optionally, select the **Duplicate from** check box and then select an existing workflow from the drop-down list.
5. Click **OK** to save your settings. The **<workflow_name>** window opens.
6. The window is divided into two panes. The Design pane on the left is where you design your workflow and contains two default elements: **Start** and **End**.

The Elements pane contains gateways and tasks that you can use in your workflow. The following elements are available:

- **Tasks** - All existing Data Warehouse ETL tasks, Data Mart ETL tasks, and Command tasks.

- **Gateways** - There are two types of gateway: **Parallel Split** and **Synchronize**. Use the Parallel Split gateway to create parallel paths. This is useful, for example, if you want two or more ETL tasks to run in parallel.

  Use the Synchronize gateway to merge parallel paths. The workflow waits for all the Tasks that precede the gateway to complete before continuing the flow.

7. To design your workflow:
   a. Drag the desired workflow elements from the Elements pane (on the right) to the Design pane (on the left).
   b. Arrange the elements in the order that you want them to run.
   c. Connect the elements to each other by dragging the connector from the gray dot (that appears on the right of an element when you hover the mouse cursor over it) to the
target element. When a blue outline appear around the target element, release the mouse button.

d. Optionally add error paths to the workflow. The workflow will follow the error path if a task encounters an error. For example, if an error occurs with a Data Mart ETL task, you may want to run another Data Mart ETL task in its place.

To add an error path, hover your mouse cursor over the task element. A red dot will appear below the element. Drag the connector from the red dot to the target element, as shown below.

![Diagram showing error path](image)

**Note** Connecting two error paths to the same task should be avoided as the workflow will fail if the task tries to run twice.

**Continuing a Workflow in the Event of Parallel Task Failure**

In a workflow, all task elements have an error port. This allows you to change the course of the workflow in the event of a task failure, as described in Step *Creating Workflows* above. Similar to Task elements, the *Synchronize* gateway also has an error port which can be used to reroute the workflow if any of the tasks between the *Parallel Split* and *Synchronize* gateways should fail.

By default, a workflow will end with an error if one or more parallel tasks do not complete successfully. However, in certain cases you may want the workflow to continue, even if one or more of the parallel tasks failed.

To do this, you need to connect the error port of the relevant task(s) directly to the *Synchronize* gateway. You can also design the workflow so that it follows the path leading from the *Synchronize* error port, instead of continuing its normal flow.

In the example below, the error port of the *MyCommandTask* is connected to the *Synchronize* gateway, meaning that even if *MyCommandTask* task fails, the workflow will continue. However, if the *MyCommandTask* task fails, the workflow will not proceed...
Validating Workflows

It is strongly recommended to validate your workflow before running it. This will prevent errors from occurring during runtime due to an invalid workflow.

Workflow rules include:
- All elements must be connected to each other
- A workflow must contain Start and End elements and at least one task.
- A workflow cannot contain a Parallel Split gateway without a Synchronize gateway and vice versa.
- Data warehouse ETL tasks cannot run in parallel with data mart ETL tasks.
- Data warehouse ETL tasks that update the same tables cannot run in parallel.
- A workflow cannot contain a Parallel Split gateway without a Synchronize gateway and vice versa.
- The execution order of elements must be sequential and not cyclic. For example an element cannot loop back to an element that precedes it the execution order.

To validate your workflow:
- Click the Validate Flow toolbar button.

If the workflow is valid, a "<workflow_name> is valid." message will be appear at the top of the window. If the workflow is not valid, a message describing the problems will appear instead.

Managing Workflows

The table below describes the options available for managing workflows.
To | Do This
--- | ---
Delete a Workflow | In Monitor view, select the workflow in the **Task** column and then click the **Delete Workflow** toolbar button.
Edit a Workflow | In Monitor view, either double-click the workflow you want to edit or select the workflow and click the **Open** toolbar button. Continue from **Creating Workflows**.
Delete an element in workflow | Either right-click the element and select **Delete** or select the element and then click the **Delete** toolbar button.
Reset the workflow view | Click the reset button to the right of the slider at the top of the window.
Zoom in to/zoom out of the workflow | Move the slider at the top of the window to the left or right as required.

Running and Monitoring Workflows
You can either schedule workflows as described in **Scheduling Tasks** or run them manually using the **Run** toolbar button. The **Run** toolbar button appears both in the main Monitor view and in the workflow design window. Note that when you run a workflow from the workflow design window, a new **Monitor** tab is added to the window and the view automatically switches to the **Monitor** tab.

You can monitor the workflow either in the **Monitor** tab or in the **Progress Status** tab. During runtime, the workflow elements fill with blue providing a graphic indication of progress. If a task encounters an error, the task element will appear with red fill instead of blue.

The figure **Example Compose Workflow** shows a workflow containing three data warehouse ETL tasks and one data mart ETL task. In the workflow, the data mart ETL task will be run only after the completion of the parallel data warehouse ETL tasks.

**Figure 11.1 | Example Compose Workflow**

![Example Compose Workflow Diagram](image)
Monitoring and Controlling Replicate Tasks

Before you can create a Compose project, you need to define a Replicate task that replicates the relevant source tables from the source database to the landing area. You can define a different task for each project or the same task can serve several projects.

Monitoring and controlling Replicate tasks from within Compose involves the following steps:

» **Step 1**: Configure Attunity Compose to connect to the Attunity Replicate machine.

» **Step 2**: Add the Replicate task name to the data source settings.

» **Step 3**: Monitor and control the Replicate task.

**Step 1: Configure Attunity Compose to connect to the Attunity Replicate machine**

1. Either click the **Replicate Server Settings** link below the **Associate with Replicate task** field in the **New Data Source** dialog box.
   - OR-
     From the **Management** drop-down menu in the main toolbar, select **Replicate Server Settings**.
     The **Replicate Server Settings** dialog box opens.

2. In the **Server Name** field, enter the IP address or host name of the Attunity Replicate machine.

3. Optionally, change the default port (443). You should only change the default port if you are certain that a different SSL port is being used.

4. Enter your **User Name** and **Password** for connecting to the Attunity Replicate machine.

5. Click **Test Connection** and then click **OK** if the connection is successfully verified.

**Step 2: Add the Replicate task name to the data source settings**

1. Open the **New Data Source** or **Edit Data Source: <Name>** dialog box as described in **Adding Data Sources in Compose**.

2. Click the browse button to the right of the **Associate with Replicate Task** field.
   The **Select Replicate Task** window opens.

3. Select the task that is replicating the source tables to the landing area and then click **OK**.

4. Make sure that the remaining data source fields are configured correctly (as described in **Adding Data Sources in Compose**) and then click **OK** again to save the data source settings.
   The Replicate task is immediately added to the Compose monitor.
Step 3: Monitor and control the Replicate task

Figure Monitoring and Controlling Replicate Tasks shows how the Replicate task appears in the Compose Monitor. You can stop and start the Replicate task using the **Abort** and **Run** toolbar buttons.

**Note** If a task is stopped from within Replicate, the task status in Compose will be "Completed" instead of "Aborted".

You can also define notifications for the task and add the task to a workflow. For more information, see Defining Notifications Rules and Workflows respectively.

The monitor provides various information about the task. For details, see Viewing Information in the Monitor.

**Figure 11.2 | Replicate Task in the Compose Monitor**
Attunity Compose management options can be accessed from the Management menu located at the top of the Compose main page.

The following management options are available:

- Register or request a License to work with Attunity Compose.
- Set the logging level for the Attunity Compose Server.
- Define and manage mail settings for notifications
- Define and manage Java server settings for running Compose tasks remotely
- Define and manage Replicate server settings for monitoring Replicate tasks
- Define and manage user permissions

License Settings

You need to register a valid license in order to use Attunity Compose. The license file contains details such as the product expiration date, the date the license was issued, which source databases can be used, and so on.

The following sections describe how to register and view your Compose license:

- License Settings
- Viewing a License

Registering a License

This section describes how to register your Compose license. You can register the license either using the console or using a command line.

To register a license using the console:

1. Copy the license file to the computer on which Compose is installed or to any computer in your network that can be accessed from the Compose computer.
2. From the **Management** menu, select **License|Register License**.
   The **Register License** dialog box opens.

![](image1.png)

3. 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.

4. Click **Register License** to register the license. A message indicating the license was registered successfully is displayed.

**To register a license using the command line:**

1. Open a command prompt and change the working directory to:
   ```
   PRODUCT_DIR\bin
   ```
Note The default \PRODUCT\_DIR is: C:\Program Files\Attunity\Compose

2. Issue the following command:

ComposeCli.exe register_license --req @license_file|license_text

where:

» license_file is the full path to the Attunity Compose license file. Note that the path should always be preceded by the "@" symbol.

Example:

ComposeCLI.exe register_license --req @c:\Admin\Temp\lic.txt

» license_text is a string in JSON format. When specifying a JSON string any quote symbols should be escaped using a backslash (\).

Example:

ComposeCli.exe register_license --req "{"$type": "ComposeLicense", "product": "AttunityCompose", "issued_to": "qa", "issued_by": "Attunity Israel", 07-17", "hosts": "", "product_version": "2.8", "notes": "", "host_role": "", "source_db_types": "", "dwh_type": "", "number_of_dms": "0", "managed_dwh_size": "0"}", LcVLPfXvD4wY5ZyUYlasdjt0vQd1Hwk5UzT7xe5+pqhZtB1dfUUy150+7zKju7vm1kkPnz3I+L5Lb1m3FpvxIx0Frfj2LQBk1LoUXMN+v06vI+w5aMSGQw6fttUgbYohFCIOduk8="}

If the license is registered successfully, the following message will be displayed:

Compose control program completed successfully.

Otherwise, an appropriate error message will be displayed.

Viewing a License

You can view the license information in the Attunity Compose Console at any time.

To view the license information:

From the Management menu, select License|View License.

The License window opens. All of the license information is displayed in the License window.

Logging Settings

The following topics describe the server logging settings:

» Setting the Server Logging Level
» Setting Automatic Roll Over and Cleanup
» Viewing and Downloading Compose Log Files
Setting the Server Logging Level
The logging level determines what type of information is written to the log files. The log files provide information about Attunity Compose server processes.

The following logging levels are available (ordered from the lowest level to the highest level):
1. Errors Only
2. Warnings
3. Info
4. Debug
5. Detailed Debug

The higher levels always include the messages from the lower levels. Therefore, if you select Error Only, only error messages are written to the log files. However, if you select Info, informational messages, warnings, and error messages will be included. Selecting Detailed Debug writes all possible messages to the log.

You can set a global logging level for all log components or you can set a separate logging level for each component.

To set the server logging level:
1. From the Management menu, select Logs|Log Management.
   The Log Management window opens displaying the Server Log tab.
2. To set a global logging level, move the top slider to the level you want. All of the sliders for the individual modules move to the same level that you set in the main slider.
3. To set a logging level for individual Compose components, select a module and then move its slider to the desired logging level.

   **Note** Changes to the logging level take place immediately. There is no need to restart the Attunity Compose service.

4. Click OK to save your changes and close the Log Management dialog box.

Setting Automatic Roll Over and Cleanup
You can define when log files should be automatically rolled over as well as how many days to keep log files.

To set the log file roll over and cleanup options:
1. From the Management menu, select Logs|Log Management.
   The Log Management window opens.
2. Select the Log Settings tab.
3. The following options are available:
» **Enable automatic roll over:** Select this check box to determine the maximum size a log file can reach before it is rolled over. The current log file is called Compose.log and saved (older) log files are called Compose_xxxxxxxxxxxxxx.log where xxxxxxxxxxxxxxx represents a 12-digit timestamp.

» **Roll over the log if the log file is larger than (MB):** Use the counter or type in the maximum amount of megabytes for a specific log file. When the log file reaches the specified size, the old log is saved with a timestamp appended to its name and a new log file is started. The default value is 10 megabytes.

**Note** The scheduled job that checks the log size runs every five minutes. Consequently, the actual size of the log when rolled over might be larger than specified.

» **Enable automatic cleanup:** Select this check box to define the maximum number of days a log file can exist before it is deleted.

» **Delete log files that are older than (days):** Use the counter or type in the maximum number of days to keep a log file. Log files that are older than the specified number of days are automatically deleted from the system. For example, if 4 is specified, then all log files will be deleted on the fifth day. The default value is 10 days.

4. Click **OK** to save your settings and close the **Log Management** window.

Viewing and Downloading Compose Log Files
This section explains how to view and download Compose log files.

**To view Compose log files:**
1. From the **Management** menu, select **Logs|View Logs**.
   The **Log File Viewer** opens.
2. Select the log file you want to view from the list in the **Log Files** pane.
   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 display the full message of the selected row.
3. Browse through the log file using the scroll bar on the right and the navigation buttons at the top of the window.
4. 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.

**To download Compose log files:**
1. From the **Management** menu, select **Logs|View Logs**.
   The **Log File Viewer** opens.
2. From the list in the Log Files pane, select the log file you want to download.
3. Click the Download Log File button in the top right of the window.
   The log file is downloaded.

Mail Server Setting
The Mail parameters define the mail server used to send notifications.

To set the log file roll over and cleanup options:
1. From the Management menu, select Mail Settings.
   The Mail Server Settings window opens.
2. Configure the settings as follows:
   » Mail Server: Specify the outgoing mail server that will be used to send Attunity Compose notifications, for example, smtp.example.com.
   » Port: Enter the mail server port number. The default value is 25.
   » Use SSL: Select this check box to connect to the mail server using SSL.
   » Anonymous Login: Enable this to allow Attunity Compose to access the mail server without having to provide any user credentials.
   » User Name: Specify the user name for the account that will be used to send notifications.
   » Password: Specify the password for the account that will be used to send notifications.
   » 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 Mail: You this option to validate your mail server settings.
   Click Send Test Mail to open the Send Test Email dialog box.
   In the Email address for test email, enter the email address to which you want the test email to be sent and then click Send.
3. Click OK to save your settings and close the Mail Server Settings dialog box.

Running Tasks on a Remote Compose Server
You can run Compose tasks either locally (the default) or on a remote Compose server.

To run tasks on a remote server:
1. From the Management menu in the projects view, select Java Server Settings.
2. In the Java Server Settings dialog box, select Remote Server and provide the required connection details.
3. Click OK to save your settings.
Monitoring and Controlling Replicate Tasks

Before you can create a Compose project, you need to define a Replicate task that replicates the relevant source tables from the source database to the landing area. You can define a different task for each project or the same task can serve several projects.

Monitoring and controlling Replicate tasks from within Compose involves the following steps:

» **Step 1:** Configure Attunity Compose to connect to the Attunity Replicate machine.
» **Step 2:** Add the Replicate task name to the data source settings.
» **Step 3:** Monitor and control the Replicate task.

**Step 1: Configure Attunity Compose to connect to the Attunity Replicate machine**

1. Either click the [Replicate Server Settings] link below the [Associate with Replicate task] field in the [New Data Source] dialog box..

   - OR-

   From the [Management] drop-down menu in the main toolbar, select [Replicate Server Settings].

   The [Replicate Server Settings] dialog box opens.

2. In the [Server Name] field, enter the IP address or host name of the Attunity Replicate machine.

3. Optionally, change the default port (443). You should only change the default port if you are certain that a different SSL port is being used.

4. Enter your [User Name] and [Password] for connecting to the Attunity Replicate machine.

5. Click [Test Connection] and then click [OK] if the connection is successfully verified.

**Step 2: Add the Replicate task name to the data source settings**

1. Open the [New Data Source] or [Edit Data Source: <Name>] dialog box as described in [Adding Data Sources in Compose].

2. Click the browse button to the right of the [Associate with Replicate Task] field.

   The [Select Replicate Task] window opens.

3. Select the task that is replicating the source tables to the landing area and then click [OK].

4. Make sure that the remaining data source fields are configured correctly (as described in [Adding Data Sources in Compose]) and then click [OK] again to save the data source settings.

   The Replicate task is immediately added to the Compose monitor.
Step 3: Monitor and control the Replicate task

Figure Monitoring and Controlling Replicate Tasks shows how the Replicate task appears in the Compose Monitor. You can stop and start the Replicate task using the Abort and Run toolbar buttons.

Note If a task is stopped from within Replicate, the task status in Compose will be "Completed" instead of "Aborted".

You can also define notifications for the task and add the task to a workflow. For more information, see Defining Notifications Rules and Workflows respectively.

The monitor provides various information about the task. For details, see Viewing Information in the Monitor.

Figure 12.1 | Replicate Task in the Compose Monitor

Setting up User Permissions

You can grant Attunity Compose 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.
### Table 12.1 | Default User Permissions According to Role

<table>
<thead>
<tr>
<th>Permission</th>
<th>Viewer</th>
<th>Operator</th>
<th>Designer</th>
<th>Admin</th>
</tr>
</thead>
<tbody>
<tr>
<td>Monitor tasks (in Monitor view)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Perform runtime operations (e.g. start, stop, resume ETL tasks, and so on)</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Create and design projects</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Delete projects</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Manage databases</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Manage Compose (e.g. license registration, email settings, and so on)</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

**Note** The user under whose account Attunity Compose is installed will be associated with the Admin role by default.

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 your own Active Directory groups and add users to them (according to the roles that you want to allocate them).

If you create your own Active Directory groups, you need to add them to the **User Permissions** window and set their permissions as described in Managing User Permissions.

### Table 12.2 | Roles and their Corresponding Active Directory Names

<table>
<thead>
<tr>
<th>Role</th>
<th>Active Directory Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>Administrator</td>
<td>AttunityComposeAdmins</td>
</tr>
<tr>
<td>Designer</td>
<td>AttunityComposeDesigners</td>
</tr>
<tr>
<td>Operator</td>
<td>AttunityComposeOperators</td>
</tr>
<tr>
<td>Viewer</td>
<td>AttunityComposeViewers</td>
</tr>
</tbody>
</table>

Managing User Permissions

This section explains how to edit user permissions, and how to add and remove users or groups.

**To edit the user permissions:**

1. From the **Management** menu in the projects view, select **User Permissions**. The **User Permissions** window opens.
2. Adjust the permission sliders as desired.
3. Click OK to save your settings.

**To add new users or groups:**
1. From the Management menu in the projects view, select User Permissions. The User Permissions window opens.
2. Click the Add toolbar button. The Add User/Group dialog box opens.
3. Select User or Group as appropriate.
4. Enter the user or group name in the following format:
   For domain users/groups: domain\group_name or domain\user_name

   **Note**
   - Only NetBIOS domain names are supported. The NetBIOS domain name is the leftmost label in the DNS domain name. For example, entering qa.int\mike would result in a connection error whereas entering qa\mike would not.
   - Active Directory distribution groups are not supported.

   For local users/groups: computer_name\group_name or computer_name\user_name
5. Click OK to add the user/group to the User/Group list.

**To remove a user or group:**
1. From the Management menu in the projects view, select User Permissions. The User Permissions window opens.
2. Select the user/group you want to remove and then click the Remove toolbar button. The user/group is removed.
3. Click OK to save your settings.
A | Impact of DST Change on Attunity Compose

This appendix describes how Attunity Compose is affected by Daylight Saving Time (DST) and provides guidelines for handling changes brought about by DST.

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, 01:00AM becomes 02:00AM). This DST change does not impact Attunity Compose as 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, 02:00AM becomes 01:00AM). 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 Attunity Compose is stopped.

There are two 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. 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).

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 Attunity Compose, 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, Attunity generally recommends that customers first verify the impact of DST changes in their test environment.
B | Supported Platforms and Databases

In addition to listing the platforms on which Attunity Compose can be installed, this appendix also specifies which source and target database versions can be used in an Attunity Compose task.

## In this appendix:
- Supported Windows Platforms
- Supported Browsers
- Supported Data Sources
- Supported Data Warehouses

### Supported Windows Platforms
Attunity Compose can be installed on any of the following Windows platforms:

- Windows 7 (64-bit)
- Windows 10 (64-bit)
- Windows Server 2008 R2 (64-bit)
- Windows Server 2012 (64-bit)
- Windows Server 2012 R2 (64-bit)

### Supported Browsers
The Attunity Compose Web UI supports the following browsers:

- Internet Explorer 11 and above
- Chrome (always updates itself to the latest version)
- Firefox (always updates itself to the latest version)

### Supported Data Sources
Any data source supported by Attunity Replicate can be used as a data source in Attunity Compose. When using an Attunity Replicate data source, discovery needs to be performed on the landing area. Data sources (endpoints) that can be discovered directly from Attunity Compose are described in the table below.
For more information on discovering data sources, see Discovering the Source Database or Landing Area.

<table>
<thead>
<tr>
<th>Endpoint</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>MySQL</td>
<td>5.5, 5.6, 5.7</td>
</tr>
<tr>
<td>Oracle</td>
<td>11.x and 12.x</td>
</tr>
</tbody>
</table>

**Note:** All Oracle editions are supported.

<table>
<thead>
<tr>
<th>Endpoint</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM DB2</td>
<td>9.7 - 10.5</td>
</tr>
<tr>
<td>Teradata Database</td>
<td>14.10, 15.0 and 15.10</td>
</tr>
<tr>
<td>Amazon Redshift</td>
<td>N/A</td>
</tr>
</tbody>
</table>

**Supported Data Warehouses**

The table below lists the target endpoint versions supported by Attunity Compose.

<table>
<thead>
<tr>
<th>Data Warehouse</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle</td>
<td>11.x and 12.x</td>
</tr>
</tbody>
</table>

**Note** All Oracle editions are supported.

<table>
<thead>
<tr>
<th>Data Warehouse</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>Teradata Database</td>
<td>14.10, 15.0 and 15.10</td>
</tr>
<tr>
<td>Amazon Redshift</td>
<td>N/A</td>
</tr>
</tbody>
</table>
Cron expressions can be used to schedule a Compose task. This appendix describes the Cron format used in Compose (Quartz), provides a description of the special characters that can be used in an expression and ends with some examples of Cron usage.

In this appendix:
Cron Format
Special Characters
Usage Examples

Cron Format
A cron expression is a string comprised of five fields separated by a white space. Fields can contain any of the allowed values, along with various combinations of the allowed special characters for that field. The fields are described in the table below.

<table>
<thead>
<tr>
<th>Field Name</th>
<th>Mandatory</th>
<th>Allowed Values</th>
<th>Allowed Special Characters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Seconds</td>
<td>✓</td>
<td>0-59</td>
<td>, - * /</td>
</tr>
<tr>
<td>Minutes</td>
<td>✓</td>
<td>0-59</td>
<td>, - * /</td>
</tr>
<tr>
<td>Hours</td>
<td>✓</td>
<td>0-23</td>
<td>, - * /</td>
</tr>
<tr>
<td>Day of month</td>
<td>✓</td>
<td>1-31</td>
<td>, - * ? / L W</td>
</tr>
<tr>
<td>Month</td>
<td>✓</td>
<td>1-12 or JAN-DEC</td>
<td>, - * /</td>
</tr>
<tr>
<td>Days of week</td>
<td>✓</td>
<td>1-7 or SUN-SAT</td>
<td>, - * ? / L #</td>
</tr>
</tbody>
</table>

Special Characters
The following special characters are supported:

*  ("all values") Used to select all values within a field. For example, * in the minute field means "every minute".

?  ("no specific value") Useful when you need to specify something in one of the two fields in which the character is allowed, but not the other. For example, if I want my task to run on a particular day of the month (say, the 10th), but don’t care what day of the week that happens to be, I would put "10" in the day-of-month field, and "?" in the day-of-week field. See the examples below for clarification.
- Used to specify ranges. For example, "10-12" in the hour field means "the hours 10, 11 and 12".

, Used to specify additional values. For example, "MON,WED,FRI" in the day-of-week field means "the days Monday, Wednesday, and Friday".

/ Used to specify increments. For example, "0/15" in the seconds field means "the seconds 0, 15, 30, and 45". And "5/15" in the seconds field means "the seconds 5, 20, 35, and 50". You can also specify '/' after the " character - in this case '/' is equivalent to having '0' before the '/'. '1/3' in the day-of-month field means "run every 3 days starting on the first day of the month".

L ("last") Has a different meaning in each of the two fields in which it is allowed. For example, the value "L" in the day-of-month field means "the last day of the month" - day 31 for January, day 28 for February on non-leap years. If used in the day-of-week field by itself, it simply means "7" or "SAT". But if used in the day-of-week field after another value, it means "the last xxx day of the month" - for example "6L" means "the last friday of the month". You can also specify an offset from the last day of the month, such as "L-3" which would mean the third-to-last day of the calendar month. When using the 'L' option, it is important not to specify lists, or ranges of values, as you'll get confusing/unexpected results.

W ("weekday") Used to specify the weekday (Monday-Friday) nearest the given day. As an example, if you were to specify "15W" as the value for the day-of-month field, the meaning is: "the nearest weekday to the 15th of the month". So if the 15th is a Saturday, the trigger will run on Friday the 14th. If the 15th is a Sunday, the trigger will run on Monday the 16th. If the 15th is a Tuesday, then it will run on Tuesday the 15th. However if you specify "1W" as the value for day-of-month, and the 1st is a Saturday, the trigger will run on Monday the 3rd, as it will not 'jump' over the boundary of a month's days. The 'W' character can only be specified when the day-of-month is a single day, not a range or list of days. ** The 'L' and 'W' characters can also be combined in the day-of-month field to yield 'LW', which translates to "last weekday of the month".

# Used to specify "the nth" XXX day of the month. For example, the value of "6#3" in the day-of-week field means "the third Friday of the month" (day 6 = Friday and "#3" = the 3rd one in the month). Other examples: "2#1" = the first Monday of the month and "4#5" = the fifth Wednesday of the month. Note that if you specify "#5" and there is not 5 of the given day-of-week in the month, then no firing will occur that month. ** The legal characters and the names of months and days of the week are not case sensitive. MON is the same as mon.

Usage Examples
<table>
<thead>
<tr>
<th>Cron Expression Example</th>
<th>Creates Trigger that Fires at</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 0 12 * * ?</td>
<td>12 pm (noon) every day</td>
</tr>
<tr>
<td>0 15 10 ? * *</td>
<td>10:15 am every day</td>
</tr>
<tr>
<td>0 15 10 * * ?</td>
<td>10:15 am every day</td>
</tr>
<tr>
<td>0 15 10 * * ? *</td>
<td>10:15 am every day</td>
</tr>
<tr>
<td>0 15 10 * * ? 2005</td>
<td>10:15 am every day during the year 2005</td>
</tr>
<tr>
<td>0 * 14 * * ?</td>
<td>Every minute starting at 2 pm and ending at 2:59 pm, every day</td>
</tr>
<tr>
<td>0 0/5 14 * * ?</td>
<td>Every 5 minutes starting at 2 pm and ending at 2:55 pm, every day</td>
</tr>
<tr>
<td>0 0/5 14,18 * * ?</td>
<td>Every 5 minutes starting at 2 pm and ending at 2:55 pm, AND fires every 5 minutes starting at 6 pm and ending at 6:55 pm, every day</td>
</tr>
<tr>
<td>0 0-5 14 * * ?</td>
<td>Every minute starting at 2 pm and ending at 2:05 pm, every day</td>
</tr>
<tr>
<td>0 10,44 14 ? 3 WED</td>
<td>2:10 pm and at 2:44 pm every Wednesday in the month of March,</td>
</tr>
<tr>
<td>0 15 10 ? * MON-FRI</td>
<td>10:15 am every Monday, Tuesday, Wednesday, Thursday and Friday</td>
</tr>
<tr>
<td>0 15 10 15 * ?</td>
<td>10:15 am on the 15th day of every month</td>
</tr>
<tr>
<td>0 15 10 L * ?</td>
<td>10:15 am on the last day of every month</td>
</tr>
<tr>
<td>0 15 10 ? * 6L</td>
<td>10:15 am on the last Friday of every month</td>
</tr>
<tr>
<td>0 15 10 ? * 6L</td>
<td>10:15 am on the last Friday of every month</td>
</tr>
<tr>
<td>0 15 10 ? * 6#3</td>
<td>10:15 am on the third Friday of every month</td>
</tr>
<tr>
<td>0 0 12 1/5 * ?</td>
<td>12 pm (noon) every 5 days every month, starting on the first day of the month</td>
</tr>
<tr>
<td>0 11 11 11 11 ?</td>
<td>Every November 11th at 11:11 am.</td>
</tr>
</tbody>
</table>
D | Attunity Compose for Amazon Redshift Tutorial

In this tutorial, you'll learn how to move data from a source database to Amazon Redshift using Attunity Replicate, and then, how to transform that data into a star schema on Amazon Redshift using Attunity Compose. Usually such a process would involve many hours of manual scripting, but thanks to Attunity Compose’s unique automation capabilities, everything should be completed in minutes!

To keep the tutorial as simple as possible, Microsoft SQL Server Express with a cut-down version of the Northwind sample database has been installed on the Attunity Compose for Amazon Redshift EC2 Instance, together with Attunity Replicate and Attunity Compose. Obviously, such a configuration is not meant to reflect a real-world scenario as databases are never as simple as Northwind and Attunity Replicate is usually not installed on the same machine as the database. Nevertheless, the concepts and procedures outlined in this tutorial can be applied to all environments, regardless of what database you are using or where it is located (on-premises or in the Cloud).

**Note**  After launching your Attunity Compose for Amazon Redshift AMI EC2 Instance you will need to contact Attunity for the required software licenses. Once contacted, we will provide you with licenses for Attunity Replicate and Attunity Compose and answer any questions you may have.

**In this tutorial:**
- **PART 1:** Satisfy the Prerequisites
- **PART 2:** Launch the Attunity Compose for Amazon Redshift AMI
- **PART 3:** Use Attunity Replicate to Move the Source Data to Amazon Redshift
- **PART 4:** Use Compose to Create a Data Mart on Amazon Redshift

**What Next?**

**PART 1: Satisfy the Prerequisites**

The following prerequisites must be met prior to performing the tasks outlined in this tutorial:

- Purchase the **Attunity Compose for Amazon Redshift BYOL** offering from Amazon Marketplace.
**Amazon Redshift Cluster:** If you haven't already done so, set up an Amazon Redshift cluster and make sure that following information about your Amazon Redshift Cluster is readily available:

- Amazon Redshift Cluster Name
- Amazon Redshift Cluster Port
- Amazon Redshift User Name and Password
- Amazon Redshift Database Name

Attunity Replicate performs the following operations on the replicated tables within Amazon Redshift:

- CRUD (Select, Insert, Update, Delete)
- Bulk Load
- Create, Alter, Drop (if required by the task's definition)

  If the user is the 'DB Owner', these permissions are in place by default. Otherwise, the user must be granted these permissions to achieve successful replication.

**Port:** Make sure that port 5439 (the Amazon Redshift Cluster port) is open for inbound connections from your Attunity Compose for Amazon Redshift AMI EC2 instance.

**Amazon S3 Bucket:** Make sure that the following information about your Amazon S3 Bucket is readily available:

- Amazon S3 Bucket Name
- Amazon S3 Bucket Region
- Amazon S3 Access Key and Secret Key

  Ensure that HTTPS Internet access is enabled to access the S3 bucket

The Attunity Replicate user must have the following permission on the Amazon S3 staging bucket:

- On the Amazon S3 Bucket itself: 'List Bucket Objects'
- On the Amazon S3 Bucket Folder: 'Read, Write, Delete of Objects'

The following is a sample policy with these permissions:

```json
{
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "s3:ListAllMyBuckets"
            ],
            "Resource": "arn:aws:s3:::*"
        }
    ]
}
```
{  
  "Effect": "Allow",  
  "Action": [  
    "s3:ListBucket",  
    "s3:GetBucketLocation"  
  ],  
  "Resource": "arn:aws:s3:::demo-virginia"  
},  
{  
  "Effect": "Allow",  
  "Action": [  
    "s3:PutObject",  
    "s3:GetObject",  
    "s3:DeleteObject"  
  ],  
  "Resource": "arn:aws:s3:::demo-virginia/*:  
}  
]  

Proceed to PART 2: Launch the Attunity Compose for Amazon Redshift AMI.

PART 2: Launch the Attunity Compose for Amazon Redshift AMI

The Attunity Compose for Amazon Redshift AMI comes pre-installed with everything you need to successfully complete this tutorial: Attunity Replicate, Attunity Compose and Microsoft SQL Server (Express Edition). This part of the tutorial will guide you through the steps required to successfully launch an Attunity Compose for Amazon Redshift AMI EC2 Instance.

2. Click Continue.
3. As with any EC2 Instance, a Key Pair is required in order to generate a Windows password. The password will enable you to log in to your Attunity Compose for Amazon Redshift EC2 instance remotely (using Remote Desktop or a similar tool).
4. If you already have a Key Pair, skip to Step 7.
5. Scroll down to the bottom of the Key Pair section. Click Key Pair and then follow the instructions to create a Key Pair.
6. After your Key Pair is created, you will be redirected to the **Launch on EC2** page.

7. Click **Accept Terms & Launch with 1-Click**.

8. A subscription confirmation message will be displayed. Close the message and wait for your EC2 instance to be ready. This may take a few minutes.

9. Once the EC2 instance is running, select **Manage in AWS Console**.
10. The AWS Console will open, showing your EC2 instance.

11. Right-click on the instance and select **Get Windows Password**. If a "Your password is not ready" message is displayed, wait a few minutes and then try again.

12. The **Retrieve Default Windows Administrator Password** window opens.
13. Browse to your Key Pair file and then click **Decrypt Password**.

14. The information required to connect remotely to your Attunity Compose for Amazon Redshift EC2 Instance will be displayed at the bottom of the page. Make a note of the **Public DNS** (i.e. the EC2 IP address). You will be prompted for this later when connecting to your Amazon Redshift EC2 Instance.
Proceed to **PART 3: Use Attunity Replicate to Move the Source Data to Amazon Redshift.**

**PART 3: Use Attunity Replicate to Move the Source Data to Amazon Redshift**

Before you can use Attunity Compose to create the data warehouse tables, you first need to move the source data to the "landing area" in your Amazon Redshift data warehouse. This is done using Attunity Replicate, Attunity's high-performance data replication and loading software.

This process consists of the following steps:

**Step 1: Register your Attunity Replicate License**

1. Double-click the desktop shortcut to open Attunity Replicate.
   
   You will be prompted for your user name and password.

2. Enter the credentials for your Attunity Compose for Amazon Redshift EC2 Instance.
   
   The Attunity Replicate console opens.

3. Click the **Register License** link at the top of the console.

4. In the **Register License** window, click **Load** and browse to the license file that you received from Attunity.
5. Click **Register License** to register the license.
6. A message confirming that the license was registered successfully is displayed. Close the **Register License** window.

**Step 2: Open Attunity Replicate and Add a New Task**
1. Double-click the desktop shortcut to open Attunity Replicate. You will be prompted for your user name and password.
2. Enter the credentials for your Attunity Compose for Amazon Redshift EC2 Instance.
3. Click the **New Task** toolbar button
4. In the **New Task** dialog box, specify a unique **Name**
5. For this tutorial, make sure that the **Apply Changes** and **Store Changes** options are not selected and then click **OK**.
   In a real world scenario, you could easily set up the task to capture any changes to the source tables and apply them to the target tables.

**Step 3: Define your Data Source**
1. Click the **Manage Endpoint Connections** toolbar button.
2. In the **Manage Endpoint Connections** dialog box, click **New Endpoint Connection**.
3. In the **Name** field, specify a unique name (e.g. SQLServer).
4. Select **Source** as the endpoint **Role**.
5. From the **Type** drop-down list, select **Microsoft SQL Server**.
6. In the **Server Name** field, enter localhost.
7. Select the **SQL Server authentication** option.
8. In the **User name** field, enter ComposeDemo.
9. In the **Password** field, enter ComposeDemo (the same as the user name).
10. Click the Browse button to the right of the **Database name** field and select the Northwind Source database.
11. Click **Test Connection** to verify the connection settings and then click **Save** to save your settings.

**Step 4: Define your Data Target**
1. Click **New Endpoint Connection**.
2. In the **Name** field, specify a unique name (e.g. AmazonRedshift).
3. Select **Target** as the endpoint **Role**.
4. From the **Type** drop-down list, select **Amazon Redshift**.
5. In the **Amazon Redshift target** section, enter the information required to connect to your Amazon Redshift cluster.
6. In the **Amazon S3 staging** section, enter the information required to connect to your Amazon S3 bucket.
7. Click **Test Connection** to verify the connection settings and then click **Save** to save
your settings.

8. Click **Close** to close the **Manage Endpoint Connections** dialog box.

**Step 5: Add your Source and Target to the Task Definition**
1. Drag your source endpoint (Microsoft SQL Server) from the **Endpoints** panel on the left of the console to the **Drop source endpoint here area** in the diagram on the right.
2. Drag your target endpoint (Amazon Redshift) from the **Endpoints** panel on the left of the console to the **Drop target endpoint here area** in the diagram on the right.

**Step 6: Select the Source Tables**
1. Click the **Table Selection** button in the toolbar on the right. In the **Select Tables** dialog box, click the **Search** button to list all of the tables.
2. Click the button to select all of the tables and then click **OK** to save your selection and close the **Select Tables** dialog box.

**Step 7: Set the Target Schema**
1. Click the **Task Settings** toolbar button.
2. In the **Target Metadata** tab’s **Target table schema** field, specify the schema in which you want the target tables to be created. You will need to specify the schema name later when configuring the data source in Attunity Compose. The schema will be created automatically if it does not already exist.

**Step 8: Run the Task**
1. Click the **Run** toolbar button to run the task.
2. The console switches to **Monitor** view. Verify that the task completed successfully (ignore the "Truncation" warning) and then proceed to **Stage 6: Use Compose to Create a Data Mart on Amazon Redshift**.
Proceed to **PART 4: Use Attunity Compose to Create a Data Mart on Amazon Redshift.**

**PART 4: Use Compose to Create a Data Mart on Amazon Redshift**

In Part 3, you learned how to define an Attunity Replicate task that replicates the data from your source tables to the landing area in Amazon Redshift. In this part of the tutorial, you'll learn how to use Attunity Compose to create a data mart on Amazon Redshift. This process consists of the following steps:

**Step 1: Register your Attunity Compose License**

1. Double-click the desktop shortcut to open Attunity Compose. You will be prompted for your user name and password.
2. Enter the credentials for your Attunity Compose for Amazon Redshift EC2 Instance. The Attunity Compose console opens.
3. Click the **Register License** link in the middle of the console.
4. In the **Register License** window, click *Load* and browse to the license file that you received from Attunity.
5. Click **Register License** to register the license.
6. A message confirming that the license was registered successfully is displayed. Close the **Register License** window.

**Step 2: Add a New Compose Project**

1. Open Attunity Compose by double-clicking the Attunity Compose desktop shortcut.
2. Click the **New Project** toolbar button.
3. Enter a name for your project and then click **OK**.

**Step 3: Configure Connections to your Data Warehouse and Data Source**

1. Click the **Manage** button in the bottom left of the **DATABASES** panel. The **Manage Databases** dialog box opens.
2. Click **New**. The **New Data Warehouse** dialog box opens.
3. In the **Name** field, specify a display name for your data warehouse.
4. From the **Type** drop-down list, select Amazon Redshift.
5. In the **Server Name** field, specify the IP address of your Amazon Redshift cluster.
6. Leave the default port (5439).
7. In the **User Name** and **Password** fields, enter your credentials for logging in to your Amazon Redshift cluster.
8. In the **Database Name** field, specify the name of the database specified in the Amazon Redshift target settings in the Attunity Replicate task.
9. In the **Data Warehouse Schema** field, either use the browse button to select an existing schema or type the name of the schema in which you want the data warehouse tables to be created. Non-existing schemas will be created automatically.

10. In the **Data Mart Schema** field, either use the browse button to select an existing schema or type the name of the schema in which you want the data mart tables to be created. It is recommended to use different schemas for the data warehouse and data mart tables. Non-existing schemas will be created automatically.

11. Click **Test Connection** to verify that Compose is able to establish a connection to the specified database and then click **OK** to save your settings.

12. Click **New** again.

13. The **New Data Source** dialog box opens.

14. In the **Name** field, specify a display name for your data source.

15. From the **Content Type** drop-down list, choose **FULL_LOAD**.

16. In the **Schema name** field, enter the schema name that you specified in the **Target Metadata** tab in the Replicate task settings. For more information, see PART 3: Use Attunity Replicate to Move the Source Data to Amazon Redshift.

17. In the **Error Mart Schema Name** field, specify the schema where you want the data mart exception tables to be created. Although not covered by this tutorial, Compose lets you define Data Quality rules that can be used to cleanse or validate data. Data that is rejected by a data quality rule will be copied to tables in the specified schema.

18. Ignore the **Associate with Replicate Task** option for now. Selecting this option allows you to monitor the Attunity Replicate task from within Compose. For full details, refer to the **Attunity Compose User Guide**.

19. Select the **Source Database Connection** check box.

20. From the **Type** drop-down list, select **Microsoft SQL Server**.

21. In the **Server Name** field, specify **localhost**.

22. Leave the default port (1433).

23. In the **User Name** field enter **ComposeDemo**.

24. In the **Password** fields, enter **ComposeDemo** (the same as the user name).

25. In the **Database Name** field, use the Browse button to select **Northwind Source**.

26. In the **Schema** field, use the Browse button to select **dbo**.

27. Click **Test Connection** to verify that is able to establish a connection to the specified database and then click OK to save your settings.

28. Click **OK** to save your settings.

**Step 4: Create the Model**

As Compose is model-driven, the first thing you need to do after configuring your database connections is to create a model. There are several ways of doing this in Compose including importing the model from ERwin, but to keep things simple, we’re going to get Compose to "discover" the data source and auto-generate the model for us.
1. In the Model panel, perform the following steps to create the model for data warehouse generation:
2. From the drop-down menu in the top right corner of the MODEL panel, select Discover. The Discover dialog box opens.
3. Select the source database (i.e. the database without the "_landing" suffix).
4. The Source Table/View Selection - Name dialog box opens.
5. Make sure that Tables is selected as the Search for option (the default).
6. Click the Search button.
7. Select all the tables by clicking the button to the right of the Results list. Then click OK.
8. The Generating Model from Name window opens.
9. Wait for the "Model created successfully" message and then click Close.
10. The Source Table/View Selection - Name dialog box closes automatically.

Step 5: Create and Populate the Data Warehouse Tables

Now that our model is in place, we can proceed with creating and populating the data warehouse tables.
1. Click the Create button in the bottom right of the DATA WAREHOUSE panel.
2. The Creating Data Warehouse window opens. Wait for the Data Warehouse to be created and then click Close.
3. Click Manage.
4. The Manage ETL Sets window opens.
5. Click Generate.
6. The Generating Instructions for ETL: Name window opens. Wait for the ETL instruction set to be generated and then click Close.
7. Click Run.
8. The Manage ETL Sets window switches to Monitor view and starts to populate the Data Warehouse with data.
9. Wait for the Data Warehouse to be populated and then close the Manage ETL Sets window.

**Step 6: Create the Data Mart and Star Schema**

In the Data Mart panel, perform the following steps to create a Data Mart with a star schema:

1. Click **New**.
2. The New Data Mart dialog box opens. Leave the default name.
3. Make sure the Start New Star Schema Wizard check box is selected. Then click **OK**.
5. Make sure that Transactional is selected as the star schema type (the default) and then click **Next**.
6. In the Facts screen, select Order Details. Then click **Next**.
7. In the Dimensions screen, you’ll notice that all of the dimensions are selected by default. Again, to keep things simple, select Customers, Employees and Products only. Then click **Next**.

© 2017 Attunity Ltd.
8. In the **Transaction Date** screen, select **OrderDate**. Then click **Finish**.

   The wizard closes and the star schema with the selected fact table and dimensions is displayed on the right of the **Manage Data Marts** window.

9. Click **Create** Tables.

10. The **Creating Data Mart: Name in Target** window opens. Wait for the "Data Mart Name has been created successfully" message to appear and then close the window.

11. Click **Generate**.

12. The **Generating Instructions for ETL: Name** window opens. Wait for the process to complete and then close the window.
13. Click **Run**.

14. The **Manage Data Marts** window switches to **Monitor** view and populates the Data Mart with data.

   Leave the **Manage Data Marts** window open in **Monitor** view for now (The two buttons at the top right of the window allow you to switch between Designer and Monitor views).

---

**Step 7: View the Data in a Pivot Table**

Compose’s built-in Pivot Table allows you to generate a test report from the data in your star schema.

1. In the **Manage Data Marts** window, click the **Pivot** toolbar button.

2. The **Select Columns for Pivot Table** window opens.

3. From the drop-down list at the top of the window, select the Pivot Table columns as follows:
   - From the **1Fct_order_details** table, select **unitprice**.
   - From the **TDMA_1Dim_customers** table, select **companyname**.
   - From the **TDMA_1Dim_products** table, select **productname**.

4. Click **OK**.

   A Pivot Table is created with your selected columns.
5. Drag the `companyname` dimension to the gray empty space on the left of the Pivot table. Then, drag the `productname` dimension to the gray empty space immediately above the Pivot table.

Now we're going to make things a little more interesting by combining a "Heatmap" with a "Sum" calculation to show the sum total of all products sold by each company.

6. Select `Heatmap` from the drop-down list below the `Customize Columns` button on the left.

7. Then, select `Sum` from the drop-down list immediately below that.

8. And finally, select `unitprice` from the bottommost drop-down list.

Your pivot table should now look like this:

![Pivot Table Image]

**What Next?**

Now that you've successfully completed the tutorial, you'll probably want to waste no time applying what you've learned to your own environment. If your source databases are located on-premises, then you will need to contact our Sales Team for an Attunity Replicate download link. If your data sources are located in the Cloud (e.g. Amazon RDS), you can simply continue using Attunity Replicate on your Attunity Compose for Amazon Redshift EC2 machine.

As you've probably guessed, this tutorial barely scratches the surface of what you can do with Attunity Replicate and Attunity Compose. For an in-depth description of the products' features and capabilities, we suggest perusing the user guides. These are available either as a PDF under the program listing in the Windows **All apps** menu, or by clicking the "Help" icon within the web console.
A

Attribute
In the Compose model, an attribute is a logical representation of a physical column in a source database (or landing area) table.

Attributes Domain
A list of all the attributes available in the Compose model. You can add, edit and delete attributes according to your data warehousing needs. The Attributes Domain also shows you which entities each attribute is used in, as a single attribute may be used in several entities.

Change Tables
Change Tables are created in the landing area when the Replicate task is defined as Full Load and Store Changes or Store Changes only. When the Store Changes replication option is enabled in the Replicate task, any changes to the source tables will be replicated to the Change Tables in the landing area. The Change Table name format comprises the original table name appended with a "__ct".

E

Entity
In the Compose model, an entity is a logical representation of a physical source database/landing area table or view.

ETL Task
In a project, the following ETL tasks can be run: - An ETL task that extracts data from the landing area, performs user-defined transformations on the data, and loads it into the data warehouse tables. - An ETL task that extracts data from the data warehouse, performs user-defined transformations on the data, and loads it into the data mart tables. Depending on the ETL task type and specific settings within Compose, only changes to the existing data will be populated or all of the data (regardless of whether any changes were made to the source).

F

Full Load
A Full Load replication task is a Replicate task that replicates all of the selected source tables to the landing area and populates them with data from the source database. When you duplicate an existing data warehouse ETL, you can set the ETL type to Full Load and Change Tables (i.e. initially extract all the data from the landing area tables and then only the changes), Full Load Only (i.e. extract all the data from the landing area tables) or Change Tables Only (i.e. extract only the changes to the landing area tables).
History
Model attributes (and their corresponding data warehouse columns) can either be defined as history Type 1 or history Type 2. When an attribute is defined as history type 1, no history of the data is kept since old data will always be overwritten with new data. When an attribute is defined as history Type 2, a new record is added each time the record is updated. This is especially useful for Slowly Changing Dimensions (SCDs). For example, defining the Address attribute in the Customers table as Type 2 would enable you to retrieve data based on the customer’s location during a certain time period. Attributes defined as history Type 1 will always exist in hub tables whereas attributes defined as history Type 2 will always exist in satellite tables.

Hub
A table in the data warehouse containing history Type 1 columns. When a column is defined as history type 1, no history of the data is kept since old data is overwritten with new data.

Landing Area
The area in the data warehouse to which the source tables are replicated. This is also the target endpoint in an Attunity Replicate task.

Lineage
A visual representation of the data flow of a particular table or column from its source to its current location. Before editing an entity or attribute, you may want to see which other entities/attributes or tables/columns will be impacted by the change. For example, removing the "Discount" attribute from a table will affect the "Total Price". Additionally, a single attribute may have multiple names depending on its location.

Model
The business information model of an enterprise. Usually an ERD (Entity-Relationship Diagram), the model should contain all of the information needed to create the data warehouse. Models can be imported from ERwin or generated automatically by discovering (otherwise known as reverse engineering) the source database or landing area.

Relationship
Similar to a foreign key, a relationship "attribute" is a special type of attribute that points to another entity in the same model.

Satellite
A table in the data warehouse containing history Type 2 columns. When a column is defined as history type 1, a new record is added whenever a record is updated (instead the existing record being overwritten). Satellite tables also contain two additional columns: FD (From Date) and TD (To Date). For old records, these columns show the dates between which a
particular record was current (i.e. before a new record rendered it obsolete). The TD column will only contain a date if the record has been succeeded by a newer record. In Compose, you can set a satellite number (1 and above) for attributes in the model. This is a good way of ensuring that similar attributes (or columns in the data warehouse) appear in the same satellite table. For example, setting the same satellite number for the "Total" and "Discount" attributes ensures that both attributes will be included in the same satellite table.