# Contents

## Preface

<table>
<thead>
<tr>
<th>Audience</th>
<th>xii</th>
</tr>
</thead>
<tbody>
<tr>
<td>Documentation Accessibility</td>
<td>xii</td>
</tr>
<tr>
<td>Related Documents</td>
<td>xii</td>
</tr>
<tr>
<td>Conventions</td>
<td>xiii</td>
</tr>
</tbody>
</table>

## Part I  Introducing Oracle Business Intelligence Cloud Service

### 1  Getting Started with Data Preparation in Oracle Business Intelligence Cloud Service

<table>
<thead>
<tr>
<th>About Oracle BI Cloud Service</th>
<th>1-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accessing Oracle BI Cloud Service</td>
<td>1-2</td>
</tr>
<tr>
<td>How Does Oracle BI Cloud Service Integrate with Oracle Database Cloud Service?</td>
<td>1-2</td>
</tr>
<tr>
<td>Before You Begin with Oracle BI Cloud Service</td>
<td>1-3</td>
</tr>
<tr>
<td>Typical Workflow for Administrators of Oracle BI Cloud Service</td>
<td>1-3</td>
</tr>
<tr>
<td>About Oracle BI Cloud Service Roles and Users</td>
<td>1-6</td>
</tr>
<tr>
<td>How to Begin Preparing Data in Oracle BI Cloud Service</td>
<td>1-6</td>
</tr>
<tr>
<td>Top Tasks for Oracle BI Cloud Service</td>
<td>1-7</td>
</tr>
<tr>
<td>Top Tasks for Data Loading</td>
<td>1-7</td>
</tr>
<tr>
<td>Top Tasks for Data Modeling</td>
<td>1-7</td>
</tr>
<tr>
<td>Top Tasks for Managing</td>
<td>1-8</td>
</tr>
</tbody>
</table>

## Part II  Loading Data for Oracle Business Intelligence Cloud Service

### 2  Overview to Getting Your Data to the Cloud

<table>
<thead>
<tr>
<th>Uploading Data to the Cloud</th>
<th>2-1</th>
</tr>
</thead>
<tbody>
<tr>
<td>About Data Sync</td>
<td>2-2</td>
</tr>
<tr>
<td>Setting Up Data Sync for the First Time</td>
<td>2-3</td>
</tr>
<tr>
<td>Giving Users Permissions to Upload Data with Data Sync</td>
<td>2-4</td>
</tr>
</tbody>
</table>
3 Loading Data from Files Using Data Sync

Typical Workflow for Loading Data from Files Using Data Sync 3-1
About Data File Requirements 3-2
About Data Sets 3-3
Setting Up Data Loads from CSV or XLSX Files Using Data Sync 3-3
Maintaining File Setup Data 3-6
Loading Data Using Data Sync 3-6
Refreshing Data Regularly 3-7
Monitoring Data Loads 3-7
Reviewing Load Strategies 3-8

4 Loading Data from Relational Tables

Typical Workflow for Loading Data from Tables 4-1
Setting Up Data Loads from Tables Using Data Sync 4-2
Overriding a Data Load from a Table 4-3
Loading Data from Tables Using a SQL Query 4-4
Loading Relational Tables Using SQL Developer 4-4

5 Setting Up Data Loads From OTBI Using Data Sync

Typical Workflow for Loading Data from OTBI 5-1
About Loading Data from OTBI Data Sources 5-2
Specifying Connection Details for OTBI Data 5-2
Setting Up Data Loads from OTBI Folders or Subject Areas 5-2
Setting Up Data Loads from Folders Within OTBI Subject Areas 5-4
Setting Up Data Loads from OTBI Using Day-based Partitions 5-5

6 Setting Up Data Loads from JDBC Data Sources Using Data Sync

Typical Workflow for Loading Data from JDBC Data Sources 6-1
About Loading Data from JDBC Data Sources 6-2
Specifying Connection Details for Generic JDBC Sources 6-4
Setting Up a Data Load from a JDBC Data Source Using Metadata Objects 6-5
Setting Up a Data Load from a JDBC Data Source Using a Query 6-6
Specifying Connection Details for NetSuite Data 6-7

7 Setting Up Data Loads From Oracle Service Cloud (RightNow)

Typical Workflow for Loading Data from Oracle Service Cloud (RightNow) 7-1
About Loading Data From Oracle Service Cloud (RightNow) 7-2
Specifying Connection Details for Oracle Service Cloud (RightNow) 7-4
Setting Up A Data Load From An Oracle Service Cloud (RightNow) Report 7-4
Setting Up a Data Load From Oracle Service Cloud (RightNow) Using a ROQL Query 7-5
Setting Up a Data Load From Oracle Service Cloud (RightNow) Using a Metadata Query 7-8
Setting Up a Data Load From Oracle Service Cloud (RightNow) Using Named IDs 7-8

8 Automating Loading Data Using Programmatic Methods

About the Oracle BI Cloud Service REST API 8-1
About the Oracle Database Cloud Service REST API 8-2
About PL/SQL Database Scripts 8-3

9 Transforming Your Data

Typical Workflow for Transforming Data Using Data Sync 9-1
About Transforming Your Data 9-2
Transforming Your Data With Default Values, Conversions, and Calculations 9-2
Transforming Your Data With New Target Columns 9-3
Transforming Your Data Using Surrogate Keys 9-4
Transforming Your Data Using Joins 9-4
Tracking Information About Your Data 9-6
Manipulating Your Data Before And After Data Loads 9-6

Part III Modeling Data

10 Understanding Data Modeling

About Modeling Data 10-1
Planning a Data Model 10-2
Understanding Data Model Requirements 10-2
Components of Data Models 10-2
About Modeling Source Objects with Star Relationships 10-3
About Modeling Source Objects with Snowflake Relationships 10-4
11  Starting to Build Your Data Model

Typical Workflow for Modeling Data 11-1
Using Data Modeler 11-2
  Opening Data Modeler 11-3
  Creating a Data Model 11-3
  Using the Left Pane in Data Modeler 11-3
  Using the Right Pane in Data Modeler 11-5
  Using Action Menus 11-6
  Locking a Data Model 11-6
  Validating a Data Model 11-7
  Refreshing and Synchronizing Source Objects and Data Model Objects 11-8
  Publishing Changes to Your Data Model 11-9
  Clearing Cached Data 11-10
  Renaming a Data Model 11-11
  Connecting a Model to a Different Database 11-11
  Exporting a Data Model 11-12
  Importing a Data Model 11-12
  Deleting a Data Model 11-13

Reviewing Source Tables and Data 11-13
  Viewing Source Objects 11-14
  Previewing Data in Source Objects 11-14
Creating Source Views 11-15
  About Source Views 11-15
  Adding Your Own Source Views 11-15
  Defining Filters for Source Views 11-17
Adding Fact Tables and Dimension Tables to a Data Model 11-18
  About Fact Tables and Dimension Tables 11-18
  Creating Fact and Dimension Tables from a Single Table or View 11-19
  Creating Fact Tables Individually 11-21
  Creating Dimension Tables Individually 11-22
  Editing Fact Tables and Dimension Tables 11-23
  Adding More Columns to Fact and Dimension Tables 11-25
    Adding Columns from Another Source to a Dimension Table 11-25
Joining Tables in a Data Model 11-26
  About Joins 11-26
  Joining Fact and Dimension Tables 11-27
Creating a Time Dimension 11-27
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adding Measures and Attributes to a Data Model</td>
<td>11-28</td>
</tr>
<tr>
<td>Editing Measures and Attributes</td>
<td>11-29</td>
</tr>
<tr>
<td>Specifying Aggregation for Measures in Fact Tables</td>
<td>11-30</td>
</tr>
<tr>
<td>Creating Calculated Measures</td>
<td>11-32</td>
</tr>
<tr>
<td>About Creating Calculated Measures</td>
<td>11-33</td>
</tr>
<tr>
<td>Creating Derived Attributes</td>
<td>11-35</td>
</tr>
<tr>
<td>Creating Expressions in the Expression Editor</td>
<td>11-35</td>
</tr>
<tr>
<td>About the Expression Editor</td>
<td>11-36</td>
</tr>
<tr>
<td>Creating an Expression</td>
<td>11-37</td>
</tr>
<tr>
<td>Copying Measures and Attributes</td>
<td>11-38</td>
</tr>
<tr>
<td>Copying Model Objects</td>
<td>11-38</td>
</tr>
<tr>
<td>Defining Hierarchies and Levels for Drilling and Aggregation</td>
<td>12-1</td>
</tr>
<tr>
<td>Typical Workflow for Defining Hierarchies and Levels</td>
<td>12-1</td>
</tr>
<tr>
<td>About Hierarchies and Levels</td>
<td>12-1</td>
</tr>
<tr>
<td>Editing Hierarchies and Levels</td>
<td>12-2</td>
</tr>
<tr>
<td>Setting Dimension Table Properties for Hierarchies</td>
<td>12-3</td>
</tr>
<tr>
<td>Setting Aggregation Levels for Measures</td>
<td>12-4</td>
</tr>
<tr>
<td>About Setting Aggregation Levels for Measures</td>
<td>12-4</td>
</tr>
<tr>
<td>Securing Your Data Model</td>
<td>13-1</td>
</tr>
<tr>
<td>Typical Workflow for Securing Model Data</td>
<td>13-1</td>
</tr>
<tr>
<td>Creating Variables to Use in Expressions</td>
<td>13-1</td>
</tr>
<tr>
<td>About Variables</td>
<td>13-1</td>
</tr>
<tr>
<td>Defining Variables</td>
<td>13-2</td>
</tr>
<tr>
<td>Securing Access to Objects in the Model</td>
<td>13-3</td>
</tr>
<tr>
<td>About Permission Inheritance</td>
<td>13-4</td>
</tr>
<tr>
<td>Securing Access to Data</td>
<td>13-5</td>
</tr>
<tr>
<td>Part IV Managing Your Service</td>
<td>14-1</td>
</tr>
<tr>
<td>Managing What Users Can See and Do</td>
<td>14-1</td>
</tr>
<tr>
<td>Typical Workflow for Managing What Users See and Do</td>
<td>14-1</td>
</tr>
<tr>
<td>About Users and Roles</td>
<td>14-1</td>
</tr>
<tr>
<td>About Application Roles</td>
<td>14-2</td>
</tr>
<tr>
<td>Predefined Application Roles</td>
<td>14-3</td>
</tr>
<tr>
<td>Why Is the Administrator Application Role Important?</td>
<td>14-4</td>
</tr>
<tr>
<td>Configuring What Users Can See and Do</td>
<td>14-5</td>
</tr>
</tbody>
</table>
Getting Started with Application Roles 14-5
Assigning Application Roles to Users 14-6
Assigning Application Roles to Multiple Users Through Roles 14-7
Adding Members to Application Roles 14-8
Adding Your Own Application Roles 14-10
Deleting Application Roles 14-12
Functionality Enabled by Application Roles 14-12

15 Taking Snapshots and Restoring
Typical Workflow for Taking Snapshots and Restoring 15-1
About Snapshots 15-1
Taking Snapshots and Restoring Information 15-2
  Taking a Snapshot 15-2
  Restoring from a Snapshot 15-3
  Editing Snapshot Descriptions 15-4
  Deleting Snapshots 15-4
Downloading, Uploading, and Migrating Snapshots 15-5
  Downloading Snapshots 15-5
  Uploading Snapshots 15-6
  Migrating Snapshot Data 15-7

16 Performing Administration Tasks
Typical Workflow for Performing Administration Tasks 16-1
Understanding Administration Tools 16-2
Managing Database Connections 16-2
  About Database Connections 16-2
  Connecting to Data in an Oracle Cloud Database 16-3
  Securing Database Connections with SSL 16-4
Deleting Unused Data Sources 16-5
Uploading Data Models from Oracle BI Enterprise Edition 12c 16-6
  About Uploading Oracle BI Enterprise Edition Data Models to the Cloud 16-6
  Getting Your Data Model File Ready 16-8
  Uploading Data Models from a File (.rpd) Using Console 16-9
  Editing Data Models Uploaded to the Cloud 16-9
Managing Map Information 16-10
  Setting Up Maps for Dashboards and Analyses 16-10
  Editing Background Maps 16-13
Whitelisting Safe Domains 16-15
Managing How Content is Indexed and Searched 16-16
Part V Reference

A Frequently Asked Questions

Top FAQs for Data Loading A-2
Top FAQs for Data Modeling A-6
Top FAQs for Managing Oracle BI Cloud Service A-7

B Troubleshooting

Troubleshooting General Issues B-2
Troubleshooting Data Loading Issues B-2
Troubleshooting Data Modeling Issues B-4
Troubleshooting Administration Issues B-5

C Expression Editor Reference

Data Model Objects C-1
SQL Operators C-1
Conditional Expressions C-2
Functions C-3
    Aggregate Functions C-3
    Analytics Functions C-4
    Calendar Functions C-4
    Conversion Functions C-6
    Display Functions C-6
    Evaluate Functions C-7
    Mathematical Functions C-8
    String Functions C-9
    System Functions C-10
    Time Series Functions C-10
Data Sync Reference

Installing and Updating Data Sync

- About Required User Accounts and Security Guidelines
- About Prerequisites, Supported Databases, and JDBC Requirements
- Installing Data Sync
- Starting Data Sync for the First Time
  - Starting and Stopping Data Sync
  - Reconfiguring Data Sync from the Beginning
  - Uninstalling Data Sync
- Understanding Software Alerts in Data Sync
- Updating Data Sync

Help: About Data Sync

Help: Connections View

- Connection Details For A Target Database
- Connection Details For A Source Database
- Using Advanced Properties
- Using Refresh Dates

Help: Cross-project Current Jobs

Help: Current Jobs Dialog and History Dialog

Help: Email Configuration Dialog and Recipients Dialog

- Email Configuration Dialog
- Recipients Dialog

Help: File Data Dialog

- Importing Files

Help: Export Dialog and Import Dialog

- Before You Start
- Exporting Metadata
- Importing Metadata

Help: Job Schedules Dialog

Help: Jobs View

Help: Load Strategy Dialog

Help: Mark as Completed Dialog

Help: New Job Dialog

Help: Parameters/Execution Parameters dialog

Help: Patch Alerts Dialog

Help: Pluggable Data Sources Dialog
| Help: Pre/Post SQL Processing Dialog          | D-23 |
| Help: Project Summary Dialog                | D-25 |
| Help: Properties Dialog                     | D-25 |
| Help: Relational Data Dialog                | D-26 |
|   Using a SQL Query Override to Refine a Data Load from a Table | D-27 |
|   Loading Data from SQL                     | D-27 |
| Help: System Properties Dialog              | D-28 |
| Help: Target Option Dialog                  | D-29 |
| Help: Target Tables and Data Sets Dialog    | D-29 |
| Help: Welcome Dialog                        | D-29 |
| Help: Clearing the Cache After Uploading Data | D-30 |
| Help: Creating and Modifying Tables         | D-30 |
| Help: Consolidating Data from Multiple Sources | D-31 |
| Help: Creating and Modifying Data Sets      | D-32 |
| Help: Triggering Jobs from Other Tools      | D-33 |
| Help: Triggering One Job After Another Automatically | D-34 |
| Help: Uploading Data to Multiple Cloud Targets | D-35 |
|   Setting Up a Different Environment        | D-35 |
| Help: Column Mapping/Mapping Dialog         | D-36 |
Preface

Learn how to load and model data, manage users, and administer the service.

Topics:
- Audience
- Related Documents
- Conventions

Audience

Preparing Data in Oracle Business Intelligence Cloud Service is intended for business intelligence analysts and administrators who use Oracle BI Cloud Service:

- Administrators manage access to Oracle BI Cloud Service and perform other administrative duties such as backing up and restoring information for others.
- Business intelligence analysts load and model data and create reports for consumers. Data integration options range from self-service import to operational ETL updates. Analysts can select interactive visualizations and create advanced calculations to reveal insights in the data.
- Business intelligence consumers customize dashboard pages and work with their favorite reports. Dashboards allow consumers to quickly analyze and manage activity across their system.

Documentation Accessibility

For information about Oracle’s commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Related Documents

These related Oracle resources provide more information.
- Oracle Public Cloud
Conventions

Conventions used in this document are described in this topic.

Text Conventions

<table>
<thead>
<tr>
<th>Convention</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>boldface</td>
<td>Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.</td>
</tr>
<tr>
<td>italic</td>
<td>Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.</td>
</tr>
<tr>
<td>monospace</td>
<td>Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.</td>
</tr>
</tbody>
</table>

Videos and Images

Your company can use skins and styles to customize the look of the Oracle Business Intelligence application, dashboards, reports, and other objects. It is possible that the videos and images included in the product documentation look different than the skins and styles your company uses.

Even if your skins and styles are different than those shown in the videos and images, the product behavior and techniques shown and demonstrated are the same.
Part I

Introducing Oracle Business Intelligence Cloud Service

This part introduces you to Oracle BI Cloud Service.

Chapters:

• Getting Started with Data Preparation in Oracle Business Intelligence Cloud Service
Getting Started with Data Preparation in Oracle Business Intelligence Cloud Service

This topic describes how to get started with Oracle BI Cloud Service.

Topics:

• About Oracle BI Cloud Service
• Accessing Oracle BI Cloud Service
• How Does Oracle BI Cloud Service Integrate with Oracle Database Cloud Service?
• Before You Begin with Oracle Business Intelligence Cloud Service
• Typical Workflow for Administrators of Oracle BI Cloud Service
• How to Begin with Oracle Business Intelligence Cloud Service
• About Oracle BI Cloud Service Roles and Users
• Top Tasks for Oracle BI Cloud Service

About Oracle BI Cloud Service

Oracle BI Cloud Service is a BI platform in the cloud that makes analytics available to everyone, from the workgroup to the enterprise. With Oracle BI Cloud Service, it's easy to combine data from diverse sources and quickly create rich, interactive analytic applications and reports.

Reports built with Oracle BI Cloud Service are immediately available on mobile devices, with no additional programming. All iOS and Android devices are supported.

Preparing business data for analysis is easy with Oracle BI Cloud Service:

• Use simple ETL tools and industry standard APIs to import your curated data into the Oracle database and perform ongoing updates. Allow others to load data self-service. See Loading Data in Oracle Business Intelligence Cloud Service.
• Build data models to present data for analysis that better reflects the structure of your business. See Modeling Data in Oracle Business Intelligence Cloud Service.
• Manage what people can see and do in Oracle BI Cloud Service. Determine who can load and model data, who can create reports and dashboards from the data model, and who can build ad-hoc visualizations based on the same data or any alternative data source they choose to upload. See Managing What Users Can See and Do.
• Take regular snapshots to back up the data model, content that analysts save to the catalog, and security information. See Backing Up and Restoring.

Assemble compelling analytics from your business data with dozens of interactive visualizations and automatic suggestions. To learn how to create analytics in Oracle BI
Cloud Service and share them with others, see Using Oracle Business Intelligence Cloud Service.

## Accessing Oracle BI Cloud Service

Your “Welcome to Oracle BI Cloud Service” email contains a direct link to the service. Simply click this link and sign in. Alternatively, sign in to Oracle Cloud at cloud.oracle.com and then select Oracle BI Cloud Service.

Oracle BI Cloud Service displays a product tour when you sign in for the first time. At the end of the tour, you see your Home page which has links to all the Oracle BI Cloud Service features available to you.

If you’re familiar with previous versions of the Oracle BI Cloud Service user interface and you want to use the earlier version, click Open Classic Home.

### How Does Oracle BI Cloud Service Integrate with Oracle Database Cloud Service?

You need a database to store and manage the data that you analyze in Oracle BI Cloud Service. Oracle BI Cloud Service can integrate with Oracle Database Cloud - Database Schema Service or Oracle Database Cloud Service:

- Oracle Database Cloud - Database Schema Service — Single schema-based service, included with Oracle BI Cloud Service.
Oracle BI Cloud Service is integrated with Database Schema Service so there's no extra step if you want to use this database. See Using Oracle Database Cloud - Database Schema Service.

- **Oracle Database Cloud Service** — Dedicated virtual machine with a fully configured, running Oracle Database instance.

You can configure Oracle BI Cloud Service to integrate with one or more Database Cloud Service instances. For this to work, Oracle BI Cloud Service and Database Cloud Service must be running in the same data region and your administrator must provide the connection details. See Managing Database Connections and Using Oracle Database Cloud Service.

**Note:**

When you use the Projects view in Data Sync, you can use the Post Load Processing tab to post-process your Oracle Database Cloud Service data. If you have a default Database Schema Service target, then you can't use Data Sync to post-process your data.

### Before You Begin with Oracle BI Cloud Service

It's the administrator's job to get Oracle BI Cloud Service ready for others to use.

Before you allow users to sign in to Oracle BI Cloud Service, familiarize yourself with:

- **Oracle Cloud**
  
  Create and configure your account on Oracle Cloud. See Buying a Traditional Metered Subscription to an Oracle Cloud Service or Buying a Nonmetered Subscription to an Oracle Cloud Service in Getting Started with Oracle Cloud.

- **Oracle Database Cloud Service**
  
  You need Oracle Database Cloud - Database Schema Service or Oracle Database Cloud Service to store and manage data for Oracle BI Cloud Service. See How Does Oracle BI Cloud Service Integrate with Oracle Database Cloud Service?

- **Oracle Business Intelligence Cloud Service**
  
  Set up accounts for others and assign them roles in Oracle BI Cloud Service. Set up cloud database connections so business modelers and analysts can analyze their company data. See Typical Workflow for Administrators of Oracle BI Cloud Service.

### Typical Workflow for Administrators of Oracle BI Cloud Service

Here are the common tasks to administer Oracle BI Cloud Service.
<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>More information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Learn about Oracle Cloud subscriptions</td>
<td>Oracle BI Cloud Service supports traditional metered and nonmetered subscriptions.</td>
<td>Overview of Oracle Cloud Subscriptions in Getting Started with Oracle Cloud</td>
</tr>
<tr>
<td>Get an Oracle.com account</td>
<td>You must have an Oracle.com account to subscribe to Oracle BI Cloud Service. Your Oracle.com account allows you to manage your Oracle Cloud account and provides access to a variety of online applications and resources such as Oracle Store and My Oracle Support.</td>
<td>Getting an Oracle.com Account in Getting Started with Oracle Cloud</td>
</tr>
<tr>
<td>Start paid services</td>
<td>Provide your information and request a paid service.</td>
<td>Buying a Traditional Metered Subscription to an Oracle Cloud Service or Buying a Nonmetered Subscription to an Oracle Cloud Service in Getting Started with Oracle Cloud</td>
</tr>
<tr>
<td>Activate paid services</td>
<td>Oracle provisions and activates your service. When your service is ready, you'll receive a welcome email inviting you to sign in. You receive user access details, including your user name, temporary password, and identity domain name, by email from Oracle Cloud (<a href="mailto:oraclecloudadmin_ww@oracle.com">oraclecloudadmin_ww@oracle.com</a>), with the subject Setup Complete. You are ready to go.</td>
<td>The contact person for your order is designated the primary service, account, and identity domain administrator for your organization’s Oracle BI Cloud Service. This includes both My Account and My Services administration.</td>
</tr>
<tr>
<td>Learn about administrator roles</td>
<td>If you’re the contact person for your order, you have three administrator roles: account administrator, primary service administrator, identity domain administrator.</td>
<td>As account administrator, you can: • Monitor the status of services across identity domains and data centers. • Review historical utilization data about services. • Grant and revoke access to other account administrators. As service administrator, you can: • Create and manage services. • Monitor and manage individual services. As identity domain administrator, you can: • Manage users, user accounts, and roles. If you want your administrator credentials resent to you, sign in to <a href="https://myaccount.cloud.oracle.com">https://myaccount.cloud.oracle.com</a>, click Applications, click the menu icon next to the service name, and then click Resend Administrator Credentials. This regenerates and sends the welcome email with your credentials. This option is available for 60 days after the service is provisioned.</td>
</tr>
<tr>
<td>Task</td>
<td>Description</td>
<td>More information</td>
</tr>
<tr>
<td>------</td>
<td>-------------</td>
<td>------------------</td>
</tr>
<tr>
<td><strong>Sign in for the first time</strong></td>
<td>Sign in to My Services. Click the My Services Administration URL in your welcome email and sign in using the temporary password provided in the same email. Reset your temporary password as instructed. You create services for your organization from the My Services dashboard.</td>
<td>Oracle BI Cloud Service requires access to a Oracle Database Cloud - Database Schema Service. This is a prerequisite, so you should see one or more databases listed in My Services. If you haven't set one up yet, you need to do that now.</td>
</tr>
<tr>
<td><strong>Create one or more services</strong></td>
<td>Most subscriptions entitle you to set up several independent services. This allows you to create one or more instances of Oracle BI Cloud Service based on your business needs. For example, you might want to set up two services; a service dedicated for testing and a production service. Keep in mind that services are independent: • Users can’t share their content across services. • Each service must have its own cloud database. So if you want to deploy a test and production version of Oracle BI Cloud Service, you must create two database instances.</td>
<td>In the My Services dashboard, click <strong>Create Instance</strong> and select <strong>Business Intelligence</strong>. Name your service and select a cloud database from the <strong>Associations</strong> listed. You can allocate a specific number of users to each service. The total number of users across all services can’t exceed the licensed number of users. Creating Service Instances in Getting Started with Oracle Cloud</td>
</tr>
<tr>
<td><strong>Verify a service is up and running</strong></td>
<td>After creating a service, you’ll receive a confirmation email. Click the service URL provided in the email, sign in, and confirm the service is up and running.</td>
<td>Managing Your Oracle Cloud Service in Getting Started with Oracle Cloud</td>
</tr>
<tr>
<td><strong>Learn about users and roles</strong></td>
<td>Understand about user accounts and predefined roles.</td>
<td>About Users and Roles</td>
</tr>
<tr>
<td><strong>Add and manage users</strong></td>
<td>Create accounts for your users.</td>
<td>Adding Users and Assigning Roles in Getting Started with Oracle Cloud</td>
</tr>
<tr>
<td><strong>Access the service</strong></td>
<td>Access Oracle BI Cloud Service.</td>
<td>Accessing Oracle BI Cloud Service</td>
</tr>
<tr>
<td><strong>Manage what other users can see and do</strong></td>
<td>Assign appropriate application roles to everyone using the service.</td>
<td>Managing What Users Can See and Do</td>
</tr>
<tr>
<td><strong>Create database connections</strong></td>
<td>Connect Oracle BI Cloud Service to other cloud databases so business analysts can analyze company data stored at multiple locations.</td>
<td>Managing Database Connections</td>
</tr>
<tr>
<td><strong>Monitor the service</strong></td>
<td>Check on the day-to-day operation of your service, monitor performance, and review important notifications.</td>
<td>Overview of Managing Oracle Cloud Accounts and Services in Managing and Monitoring Oracle Cloud</td>
</tr>
</tbody>
</table>
About Oracle BI Cloud Service Roles and Users

Administrators must ensure everyone’s roles and privileges are properly configured before users sign in.

Some roles are specific to Oracle BI Cloud Service and some roles apply across Oracle Cloud services:

- **Oracle Cloud Roles**
  To learn about roles and privileges that are common across Oracle Cloud services, see Oracle Cloud User Roles and Privileges in *Getting Started with Oracle Cloud*.

- **Oracle BI Cloud Service Roles**
  - **Service roles**: Several predefined cloud service roles are provisioned with Oracle BI Cloud Service. See About Users and Roles.
  - **Application roles**: Access to features inside Oracle BI Cloud Service is controlled through a set of predefined application roles. To find out which features a user can access with a particular application role, see Application Roles Predefined in Oracle BI Cloud Service and Functionality Enabled by Application Roles.

How to Begin Preparing Data in Oracle BI Cloud Service

Administrators, data loaders, and data modelers each play a part preparing business data for analysis. When your business data is ready for analysis, publish the data model so that BI content developers can start to visualize the data and share analytics with co-workers, clients, and business partners.

<table>
<thead>
<tr>
<th>Task</th>
<th>User</th>
<th>More Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sign in to Oracle BI Cloud Service as the administrator</td>
<td>Administrator</td>
<td>Accessing Oracle BI Cloud Service Typical Workflow for Administrators of Oracle BI Cloud Service</td>
</tr>
<tr>
<td>Enable other users to load and model data through application roles</td>
<td>Administrator</td>
<td>Typical Workflow for Managing What Users See and Do</td>
</tr>
<tr>
<td>Connect to Oracle Database Cloud Service data sources</td>
<td>Administrator</td>
<td>Managing Database Connections</td>
</tr>
<tr>
<td>Load data for analysis into Oracle Database Cloud Service</td>
<td>Data loaders</td>
<td>Uploading Data to the Cloud</td>
</tr>
</tbody>
</table>
Top Tasks for Oracle BI Cloud Service

In Oracle BI Cloud Service, there are top tasks for data loading, data modeling, and managing.

Tasks:
- Top Tasks for Data Loading
- Top Tasks for Data Modeling
- Top Tasks for Managing

Top Tasks for Data Loading

The top tasks for data loading are identified in this topic.
- Setting Up Data Loads from CSV or XLSX Files Using Data Sync
- Setting Up Data Loads from Tables Using Data Sync
- Setting Up Data Loads from JDBC Data Sources Using Data Sync
- Setting Up Data Loads From OTBI Using Data Sync
- Setting Up Data Loads From Oracle Service Cloud (RightNow)
- Loading Data Using Data Sync
- Transforming Your Data
- Loading Relational Tables Using SQL Developer
- Automating Loading Data Using Programmatic Methods

Top Tasks for Data Modeling

The top tasks for data modeling are identified in this topic.
- Creating a Data Model
- Reviewing Source Tables and Data
- Adding Your Own Source Views
- Creating Fact and Dimension Tables from a Single Table or View
- Creating Fact Tables Individually
- Creating Dimension Tables Individually
- Joining Fact and Dimension Tables
- Creating Calculated Measures
Top Tasks for Managing

The top tasks for managing Oracle BI Cloud Service are identified in this topic.

- Assigning Application Roles to Users
- Adding Your Own Application Roles
- Taking Snapshots
- Restoring from a Snapshot
- Managing Database Connections
- Freeing Up Storage Space
- Whitelisting Safe Domains
- Managing How Content is Indexed and Searched
- Uploading Data Models from Oracle BI Enterprise Edition 12c
Part II

Loading Data for Oracle Business Intelligence Cloud Service

This part explains how to load data that you want to model using Oracle Business Intelligence Cloud Service.

Chapters:

• Getting Your Data to the Cloud
• Loading Data from Files
• Loading Data from Relational Tables
• Setting Up Data Loads From OTBI Using Data Sync
• Setting Up Data Loads from JDBC Data Sources Using Data Sync
• Setting Up Data Loads From Oracle Service Cloud (RightNow)
• Automating Loading Data Using Programmatic Methods
• Transforming Your Data
Overview to Getting Your Data to the Cloud

This topic outlines ways to upload data for Oracle BI Cloud Service and introduces Data Sync.

Topics:
- Uploading Your Data to the Cloud
- About Data Sync
- Setting Up Data Sync for the First Time

Uploading Data to the Cloud

Upload data to the cloud so that your users can start analyzing your enterprise data.

You can upload data to tables in an Oracle Cloud database (Database Schema Service or Database Cloud Service) or to a data set in Oracle BI Cloud Service. Oracle BI Cloud Service provides a client tool named Data Sync that uploads data from files, relational tables, and OTBI (Oracle Transactional Business Intelligence) reports and subject areas, but if you prefer you can use a range of other tools and technologies to upload data.

Video

You can use any of the following tools and technologies to load data:
- Data Sync (recommended)
- Oracle Data Integrator
- Oracle SQL Developer
- Oracle SQL Workshop Data Upload Utility
- Oracle Application Express Application Data Load Utility
- REST APIs
- PL/SQL scripts

Where to find more information:
- Loading data from files, see Typical Workflow for Loading Data from Files Using Data Sync.
- Loading data database tables, see Typical Workflow for Loading Data from Tables.
- Loading data from OTBI reports or subject areas, see Typical Workflow for Loading Data from OTBI.
- Loading data from JDBC data sources, see Typical Workflow for Loading Data from JDBC Data Sources.
• Loading data from Oracle Service Cloud (RightNow), see Typical Workflow for Loading Data from Oracle Service Cloud (RightNow).

• Loading data programmatically from on-premises and cloud sources using REST APIs or PL/SQL scripts, see Automating Loading Data Using Programmatic Methods.

• Transforming data, see Transforming Your Data.

Users can also load their own data sets for ad-hoc analysis with Visual Analyzer. See Adding Your Own Data in Using Oracle Business Intelligence Cloud Service.

Comparing Data-Loading Tools

This table compares the main tools and technologies that you can use to load data.

<table>
<thead>
<tr>
<th>Functionality/Tool</th>
<th>Data Sync</th>
<th>SQL Developer</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSV and delimited files</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Excel files</td>
<td>Yes*</td>
<td>No</td>
</tr>
<tr>
<td>Post load processing</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Scheduler</td>
<td>Yes</td>
<td>Yes**</td>
</tr>
<tr>
<td>Auto-retry on failure</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Customer network uses proxy</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Oracle database source</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Other relational database source</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Target Oracle Cloud database</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Database Cloud Service</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

* XLSX only

** Operating system scheduler

Oracle BI Cloud Service integrates with Database Schema Service and Database Cloud Service.

About Data Sync

Use Data Sync to upload, and manage data. You can load data from files (CSV and XLSX), various relational sources (tables, views, SQL statements), OTBI, JDBC data sources, and Oracle Service Cloud. You can load to relational tables or data sets.

If you're loading data into either Database As A Service or an on-premises database that is configured using the ‘Oracle (Thin)’ connection type, then you can also use Data Sync to transform relational data.

Download Data Sync from Oracle Technology Network and install it locally on a Windows or UNIX machine. See Installing Data Sync.

Video

After downloading and installing Data Sync, configure your working environment by following the steps in Setting Up Data Sync for the First Time. Then, you're ready to start loading data.
Use Data Sync when you want to:

- Load data sources other than Oracle.
- Load a combination of data sources, such as CSV, XLSX, and Oracle relational files.
- Perform incremental data loads or rolling deletes.
- Perform insert-only or append strategies.
- Merge data from multiple sources.
- Transform your data (if you’re using Database As A Service or an on-premises database that is configured using the ‘Oracle (Thin)’).
- Schedule data loads. You can replace, append, and update data in tables by scheduling data loads and using the Load Strategy option of this utility.

Use Data Sync to load data from these databases:

- Oracle
- DB2
- Microsoft SQL Server
- MySQL
- Teradata
- TimesTen

Load data from generic JDBC data sources too, for example:

- Greenplum
- Hive
- Impala
- Informix
- MongoDB
- NetSuite
- PostgreSQL
- Redshift
- Salesforce
- Sybase

Setting Up Data Sync for the First Time

Install and configure Data Sync so that you can load your data.

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>More Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Download and install Data Sync</td>
<td>Download Data Sync from Oracle Technology Network and follow the installation steps.</td>
<td>Installing Data Sync</td>
</tr>
</tbody>
</table>
Giving Users Permissions to Upload Data with Data Sync

To load data using Data Sync, you need a user account with appropriate privileges.

Your Data Sync user account must also have read permissions on any source databases from which you plan to load data. This user account must be separate to any federated user accounts that you create for dashboard and report users.

Ask your administrator for the appropriate permissions. These steps describe what your Cloud service administrator needs to do.

1. If required, create an account for the Data Sync user.
   For example, sign into My Services as administrator and create a user.
2. Sign into your Cloud service as administrator.
3. Click Console, then Users and Roles.
4. Navigate to the user, click Manage Application Roles, and assign the following application roles:
   • **BI Data Load Author** - Enables Data Sync users to load data into a table.
   • **BI Advanced Content Author** - Enables Data Sync users to load data into a data set.
Setting Default Options for Data Sync

Set defaults for Data Sync to suit your business needs and optimize the way Data Sync works for you. For example, you can set up a default directory for your data files, determine how much detail gets logged, how long to keep log files, and more.

For a full list of system properties and guidance on configuring them, see Help: System Properties Dialog.

1. In Data Sync, click the Views menu, then System Properties.
2. If you plan to upload data from files (CSV or XLSX), set a Data File Root Directory.
   Specify a default location for your files, such as D:\mydatafiles.
3. If your organization uses a proxy server to route calls to external websites, configure Proxy Host and Proxy Port.
4. Set other properties or keep the default settings.

Connecting Data Sync to Your Data Target and Your Data Sources

In Data Sync, use the Sources/Targets dialog in the Connections view to specify connections details for your target database and your source databases. Data Sync loads data from these sources to the target location. If you’re loading data only from data files, for example XLSX or CSV format, then you don’t need a connection in Data Sync

- For your target database, edit the connection named TARGET and specify the details of your target Cloud database. See Specifying Connection Details for Your Cloud Service.
- If you’re loading data from a database, then specify the connection details for your database. See Specifying Connection Details for a Data Source.

If you’re loading data only from data files, then you don’t need a connection in Data Sync. Go straight to the Project view, click the File Data tab, and specify your data file details. For example, you might load from a spreadsheet or CSV file.

Specifying Connection Details for Your Cloud Service

To set up a Data Sync environment, you specify connection details for your target Cloud service.

1. In Data Sync, click Connections, then click the Sources/Targets tab.
2. In the list of connections, select TARGET.
3. In the Edit dialog, specify the following details:

<table>
<thead>
<tr>
<th>Field or Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Do not change the default name TARGET.</td>
</tr>
</tbody>
</table>
Specifying Connection Details for a Data Source

To set up a Data Sync environment, you specify connection details for your source database. If you’re only loading data from files, then you can skip this task.

For guidance on specifying connection details for specific data source types:

- For any data source, see Help: Connections View
- For OTBI sources, see Specifying Connection Details for OTBI Data
- For Oracle Service Cloud (RightNow), see Specifying Connection Details for Oracle Service Cloud (RightNow)
- For JDBC sources, see Specifying Connection Details for Generic JDBC Sources
- For NetSuite sources, see Specifying Connection Details for NetSuite Data

1. In Data Sync, click Connections, then click the Sources/Targets tab.
2. Click New to add an empty record to the list of connections.
3. Use the Edit dialog to specify the connection details.
4. Click Test Connection to make sure that the connection details are valid.
5. Click Save.
## Loading Data from Files Using Data Sync

Use Data Sync to load file-based data that you want your analysts and users to analyze.

**Topics**

- Typical Workflow for Loading Data from Files Using Data Sync
- About Data File Requirements
- About Data Sets
- Setting Up Data Loads from CSV or XLSX Files Using Data Sync
- Maintaining File Setup Data
- Loading Data Using Data Sync
- Refreshing Data Regularly
- Monitoring Data Loads
- Reviewing Load Strategies

### Typical Workflow for Loading Data from Files Using Data Sync

Here are the common tasks for loading business intelligence data from files.

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>More Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set up your Data Sync environment</td>
<td>Download and install Data Sync on a local machine, and set up your Data Sync environment.</td>
<td>Setting Up Data Sync for the First Time</td>
</tr>
<tr>
<td>Prepare your data files</td>
<td>Make sure that your data files meet the formatting requirements for Data Sync.</td>
<td>About Data File Requirements</td>
</tr>
<tr>
<td>Set up your data load</td>
<td>Register the CSV or XLSX files that you want to load, and define a load strategy for you data.</td>
<td>Setting Up Data Loads from CSV or XLSX Files Using Data Sync</td>
</tr>
<tr>
<td>Manage a data load configuration</td>
<td>Make changes to an existing data load from file configuration.</td>
<td>Maintaining File Setup Data</td>
</tr>
<tr>
<td>Load data using Data Sync</td>
<td>Use Data Sync to load data into your target Cloud database.</td>
<td>Loading Data Using Data Sync</td>
</tr>
<tr>
<td>Refresh your data regularly</td>
<td>Schedule a regular data load to refresh your data.</td>
<td>Refreshing Data Regularly</td>
</tr>
<tr>
<td>Monitor Data Loads</td>
<td>Monitor the progress of data loads and respond to issues.</td>
<td>Monitoring Data Loads</td>
</tr>
</tbody>
</table>
About Data File Requirements

Before you start to load data from files, make sure that your data files meet the requirements specified here. You can load from a single data file or multiple data files at the same time. Multiple files must have the same format.

Supported File Types

You can load Comma Separated Value (CSV) files containing most common delimiters (for example, commas, tabs), or Microsoft Excel XLSX files.

Specifying a Default File Location

In the Data Sync system properties, you use the Data File Root Directory property to specify where Data Sync looks by default for data files to import.

About XLSX File Format Requirements

• You can load one or more sheets in an XLSX file.
• You can also choose to load the whole sheet or a range of cells. For example, you might specify that the data section starts in cell D4 and ends in cell J35.
• The data types are inferred from the cell type in the XLSX file. Before you import data, set the appropriate data types in the XLSX file.
• Avoid XLSX files with many sheets, because the process that reads the XLSX files is memory intensive. If you have large files, then you might have to set a higher startup memory in the datasync.bat/.sh file.

About CSV File Format Requirements

• You can load from a single file, or multiple files at the same time as long as they are in the same format.
• You can use the Data Sync import wizard to choose from a selection of delimiter types; for example, comma, tab, pipe.
• You can specify a single character as a delimiter, such as a comma (";","), or space (" "), or a tab ("t" or "[tab]").
• You must enclose a value in double-quotes if the value contains the delimiter as part of the value.
• You must enclose a value in double-quotes if the value contains new line characters.
• You can load from a file that includes a banner, which can be ignored during load. However, the transition from header to data should predictably start from a particular line number.
• You can load a file that includes timestamps and date strings. However, only one format per file can be used. For example, if there is birth date and hire date, both need to be formatted in the same way. As an example, "yyyy-MM-dd" can be used to interpret timestamp "1968-01-01" as birth date, and "2010-06-31" as hire-date.

About Error Handling and Logging

When a file is parsed and loaded, errors can result either while reading or writing. Read-related errors most commonly occur when strings are converted to an object of
type integer, decimal, or timestamp. Errors also result from invalid formatting, for example, if an attribute contains the delimiter but is not double quoted, or a line does not have as many attributes as the header.

Write-related errors can result from insufficient length or entering null into a non-null attribute. When this type of issue is detected, the errors are logged in a file in the \log directory with the naming convention CR_<Table/File Name>_<From Connection>_<To Connection>_<timestamp>.bad. This log file contains the line number of the problem record, the record itself, and the list of problems that occurred when the file was parsed.

If invalid records are identified, you must correct the original file and rerun the process. If you don't know how to correct a record in the file at the location specified in the .bad log file, then you skip the record by adding "--" in front of the line in the file (that is, comment out the text).

### About Tracking Where Data Originates

When you load data from a file, you can track where the data originated by store the filename and line number in the target database. To do this, configure your data load on the File Data tab, then on the lower pane click File Targets, then **Column Mappings**. On the Column Mappings dialog, add two new columns. For the first new column, click the **Data Transformation** field and select FILE_NAME. For the second new column, click the **Data Transformation** field and select LINE_NUMBER.

### About Data Sets

Data Sync can load your data as a data set that Data Visualization understands.

Data sets are file–based storage objects that you can use to analyze data quickly. When you use Data Sync to load data, you can specify the target format as either ‘relational’ or ‘data set’.

### Setting Up Data Loads from CSV or XLSX Files Using Data Sync

Before you start loading data, you specify information about your data files, such as the format, which columns or cells to load, and how to handle incremental data.

**Video**

Before you start, make sure that you have prepared your data files, as specified in About Data File Requirements.

1. In the Project view, click the **File Data** tab, then click **New**.
2. Select the file or files that you want to load by doing one of the following:
   - To load from a single data file, enter the full CSV or XLSX file name into the **File Name** field or click the **File Location** field and navigate to and select the data file. For example, you might enter D:/csvFiles/AIRLINE_TRAFFIC.csv to load a specific file.
If you don't specify the full directory path in the **File Name** field, then Data Sync attempts to locate the specified file in the default file location that is set in the System Property named **Data File Root Directory**.

- To load data from multiple files at the same time, use an asterisk (*) as a wildcard in the **File Name** field, as follows:
  - To load any file that starts with a name AIRLINE_TRAFFIC, enter: `D:/csvFiles/AIRLINE_TRAFFIC*.csv`
  - To load any file that starts with a name AIRLINE_TRAFFIC, enter: `D:/csvFiles/AIRLINE_TRAFFIC*.csv`
  - To load all files that end with a .csv extension, enter: `D:/csvFiles/ *.csv`

3. Enter a unique descriptive name in the **Logical Name** field.

   For example, if you're loading from multiple data files, you might enter **My_HR_Data_Combined**.

4. Click **Next** to display the Import Options dialog.

5. Use the Import Options dialog to specify how to process your data files.

   For data in CSV format, specify:

<table>
<thead>
<tr>
<th>Field or Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Codepage</td>
<td>Select the format of the file that you're importing.</td>
</tr>
<tr>
<td>Number of lines to skip</td>
<td>Enter the number of lines to skip in the file, if any. Use this option if</td>
</tr>
<tr>
<td></td>
<td>your file has a header and the data does not start on the first line. If</td>
</tr>
<tr>
<td></td>
<td>no lines should be skipped, leave the default, 0.</td>
</tr>
<tr>
<td>First line contains headers</td>
<td>Select this option if your file contains column names in a header. Data Sync</td>
</tr>
<tr>
<td></td>
<td>parses the first line to generate column names in upper case, and truncates</td>
</tr>
<tr>
<td></td>
<td>names to 30 characters. If a data file does not contain headers, then Data</td>
</tr>
<tr>
<td></td>
<td>Sync generates default column names COLUMN_1, COLUMN_2, and so on.</td>
</tr>
<tr>
<td>Delimiter</td>
<td>Select the character that separates field values in the data file. Options</td>
</tr>
<tr>
<td></td>
<td>include Comma, Pipe, Semi-colon, Space, Tab, Tilde, or Custom. If you have a</td>
</tr>
<tr>
<td></td>
<td>custom delimiter, then select Custom and enter the single-character delimiter.</td>
</tr>
<tr>
<td>Timestamp format (Java</td>
<td>Select the format of timestamp data in your data files.</td>
</tr>
<tr>
<td>style)</td>
<td></td>
</tr>
<tr>
<td>Number of lines to be</td>
<td>Leave the default value –1 to analyze all data values when evaluating data.</td>
</tr>
<tr>
<td>sampled</td>
<td></td>
</tr>
</tbody>
</table>

   For data in XLSX format, specify:

<table>
<thead>
<tr>
<th>Field or Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Timestamp format (Java</td>
<td>Select date format that is used in the spreadsheet.</td>
</tr>
<tr>
<td>style)</td>
<td></td>
</tr>
<tr>
<td>Range of Cells</td>
<td>Use the <strong>Start</strong> field to specify the cell ID of where the data starts, for</td>
</tr>
<tr>
<td></td>
<td>example D10. Use the <strong>End</strong> field to specify the cell ID of where the data</td>
</tr>
<tr>
<td></td>
<td>ends, for example H250.</td>
</tr>
</tbody>
</table>
6. Click **Next** to display the Select Target Table dialog.

7. In the **Target Table** option section of the dialog, specify:

<table>
<thead>
<tr>
<th>Field or Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target Option</td>
<td>Specify whether to load into an existing table or create a new table. If you choose to create a new table, then specify a suitable short name or edit the default name to identify this table in the target data source.</td>
</tr>
<tr>
<td>Choose output option</td>
<td>Select <strong>Relational</strong> for analysis in enterprise dashboards and analyses. Select <strong>Data Set</strong> for analysis in Data Visualization.</td>
</tr>
<tr>
<td>Remove duplicates</td>
<td>Select this option if the source XLSX file contains duplicate records and you want Data Sync to select a distinct set of rows based on a certain attribute or set of attributes that you identify as user key columns that can enforce uniqueness and resolve duplicate records.</td>
</tr>
</tbody>
</table>

8. Click **Next** to display the New Source File: Map columns dialog.

9. Use the New Source File: Map columns dialog to verify that the data types and other configuration details are correct. For example, deselect the **Load** option next to columns that you don't want to load.

10. Specify how you want to handle incremental loads:

    a. Select the **Update Rows on Match** option next to each row.

    b. Select the **Rolling Delete** option next to one of the date fields to prune the data.

11. Use the Import File dialog to review the status message, for example **Success**.

12. On the Target Tables tab, click the **Edit** tab, and set the **Rolling Delete Days** value.

13. Define how you want to handle subsequent operations on the file, such as incremental loading.

    a. Click the **File Data** tab, then click the **File Targets** sub-tab.

    b. Click the **Load Strategy** column to display the Load Strategy dialog.

    c. At the Load Strategy dialog, click **Never delete data** and **Update Table**.

    d. At the Incremental Settings dialog, select the key column or combination of columns that uniquely identify records.

    e. Click **OK**, then click **OK** on the Message dialog prompting you to create an index.

You're now ready to start loading data using a job. Display the Jobs tab and select the job that Data Sync created for your project. Alternatively, create your own job.
Maintaining File Setup Data

After you've set up a data load from one or more CSV or XLSX files, you might want to change the load configuration. For example, you might want to change the name of the target table, or remove duplicate records when you load data.

1. In Data Sync, click the Project tab, and make sure that the correct project is selected.
2. To configure the source files, click the File Data tab to change the configuration details.
3. To configure target tables or target data sets, click the Target Tables/DataSets tab to change the configuration details.
4. To manage the column mapping for a project, click the Project Summary tab.

Loading Data Using Data Sync

After you have set up your data in Data Sync, you use a job to load the data from the data source into the target database.

Jobs load data from a data source to a data target. When you create a Project, Data Sync creates a default job for you to load your data. You can use this job or create your own job to:

• Load data once only (for example, for a full initial load).
• Load data regularly (for example, for incremental loads).

1. In Data Sync, click the Jobs tab, and make sure that your project is selected.
2. In the list of jobs, select the job that Data Sync created for you, or that you created yourself.

Data Sync creates a default job for you, named with your project name appended with Jobn. For example, if you create a Project named HCM_Data, Data Sync creates a job named HCM_Data-Job1. Alternatively, you might have created your own job.

3. Click Run Job to start the data load.
Refreshing Data Regularly

After you've set up your data load, you use a job to load the data from the data source into the target database. You can use Data Sync to refresh data regularly by scheduling a job.

**Note:**
Before you start, on the Jobs sub-tab, configure a job. You can either use the default job that Data Sync created for you based on the current project name, or create your own job.

1. In Data Sync, click the Jobs tab, and make sure that your project is selected.
2. On the Job Schedules sub-tab, click New to display the New Schedule dialog.
3. Use the Name field to identify the data that you're loading.
4. In the Job list, select the Job that Data Sync created for you, or that you created yourself.
5. Use the Recurrence Pattern area to specify when and how regularly you want to load the data.
6. Specify a Start date and End date.
7. Click Save.
   At the specified date and time, the data load will start.

Monitoring Data Loads

When you load data, you use Data Sync to monitor progress and respond to loading issues.

1. In Data Sync, click the Jobs tab, and make sure that your project is selected.
2. Click the **Current Jobs** tab to monitor in-progress data loads.

   For example, review the **Run Status** and **Status Description** fields. Click **Abort** to stop a data load, or **Restart** to re-try a data load after making corrections or changes in Data Sync.

   Use the Tasks and Task Details sub-tabs for more detailed information.

3. Click the **History** tab to monitor completed data loads.

4. Click the **Tasks** sub-tab to drill into the data load details.

**Tip**: To monitor incomplete data loading jobs for all projects, use the Cross-project Current Jobs dialog. This dialog is displayed in Data Sync when you click the server status icon in the top right-hand corner of the screen.

### Reviewing Load Strategies

You review an existing load strategy for a project to see how Data Sync is loading data so that you can make changes if required.

1. Make sure that your project is selected.

2. In the Project view, select the appropriate tab for the type of source data being loaded.

   For example, display the **Relational Data** tab, the **File Data** tab, or the **Pluggable Source Data** tab.

3. In the list of sources defined for the project, select the one that you want to review.

4. Display the Load Strategy dialog:

   For relational or pluggable data sources, the **Load Strategy** option is on the Edit sub-tab.

   For file data sources, the **Load Strategy** option is on the **Load Strategy** column on the File Data\File Targets sub-tab.
5. In the Load Strategy dialog, review the settings and make changes if required.
4

Loading Data from Relational Tables

This topic describes how to load data from relational tables.

Video

Topics:

• Typical Workflow for Loading Data from Tables
• Setting Up Data Loads from Tables Using Data Sync
• Overriding a Data Load from a Table
• Loading Data from Tables Using a SQL Query
• Loading Relational Tables Using SQL Developer

Typical Workflow for Loading Data from Tables

Here are the common tasks for loading data from database tables.

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>More Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Get Data Sync up and running</td>
<td>Download and install Data Sync on a local machine, and set up the Data Sync environment.</td>
<td>Setting Up Data Sync for the First Time</td>
</tr>
<tr>
<td>Register your relational data source</td>
<td>Specify the connection details of your relational database.</td>
<td>Connecting Data Sync to Your Data Target and Your Data Sources</td>
</tr>
<tr>
<td>Set up your data load</td>
<td>Register the tables that you want to load, and define a load strategy for each table.</td>
<td>Setting Up Data Loads from Tables Using Data Sync</td>
</tr>
<tr>
<td>Load data using Data Sync</td>
<td>Use Data Sync to load data into your target Cloud database.</td>
<td>Loading Data Using Data Sync</td>
</tr>
<tr>
<td>Load data using a SQL query</td>
<td>Use Data Sync to execute a SQL query to load data into your target Cloud database.</td>
<td>Loading Data from Tables Using a SQL Query</td>
</tr>
<tr>
<td>Load data using a SQL override</td>
<td>Use Data Sync to load data but override the load using a SQL command.</td>
<td>Overriding a Data Load from a Table</td>
</tr>
<tr>
<td>Refresh your data regularly</td>
<td>Schedule a regular data load to refresh your data.</td>
<td>Refreshing Data Regularly</td>
</tr>
<tr>
<td>Monitor Data Loads</td>
<td>Monitor the progress of data loads and respond to issues.</td>
<td>Monitoring Data Loads</td>
</tr>
</tbody>
</table>
Setting Up Data Loads from Tables Using Data Sync

Before you start loading data, you specify which columns to load and how to handle incremental data.

Video

You can import table definitions to load from using any of the defined relational connections. Supported data types include CHAR, VARCHAR, TIMESTAMP, DATE, NUMBER(n), NUMBER(m,n), CLOB, and BLOB. If a source table has columns with any other data type, those columns are imported with an UNKNOWN data type, are marked as inactive, and aren't included when data is copied.

1. In Data Sync, in the Project view, click the **Relational Data** tab.

2. Click **Data From Table**.

3. In the Import Tables into [Project] dialog, select the connection in the **Data Sources** list.

4. Select the **Import** option next to each table that you want to load.

   Use the **Table Name Filter** field to narrow the list of tables displayed:

   - Enter `CONTACT` to find an exact match in the database for a table named "CONTACT".
   - Enter `CONTACT*` or `CONTACT%` to find all tables in the database whose name starts with CONTACT.
   - Enter `*CONTACT*` or `%CONTACT%` to find all tables in the database whose name contains CONTACT.

5. Click **Import Tables** to register the source tables and create entries with the same name for target tables.

   Don't rename tables. Data Sync assumes that the source table name and target table name are the same. If you want to use a different target table name, consider using queries as a source.

   By default, all table attributes are copied. If you want to exclude columns (for example, because they are not needed for analysis or contain sensitive information), then select the table in the **Target Tables** tab, select the **Table Columns** sub-tab, and click the **Inactive** option for the column or columns. If you deactivate a column, then make sure that you inspect the index definitions that might reference inactive columns. Any index that refers to an inactive or deleted column definition is dropped, but is not created. If you would like to deactivate the indexes that may refer to inactive or deleted columns, then right-click the column and select the **Identify and inactivate invalid indexes** option. This marks any indexes that refer to inactive columns inactive as well.

6. Inspect the column Attributes:

   a. Click the Target Tables tab, then click the Table Columns tab.
   b. Deselect any columns that are not needed for analysis.

7. For each table, define a strategy.

   a. In the Load Strategy column, click the Load Strategy icon.
b. Use the Load Strategy dialog to specify how to process data.

c. Use the Incremental Settings dialog to select a user-key and a date timestamp column for the incremental processing.

d. Save the details.

You’re now ready to start loading data using a job. Display the Jobs tab and select the job that Data Sync created for your project. Alternatively, create your own job.

Overriding a Data Load from a Table

In Data Sync, you can limit the amount of data that is loaded from a source table using a SQL query override.

By default, all data from a source table is copied to the target database. To limit the data loaded, provide a SQL query with a suitable WHERE clause.

1. In Data Sync, in the Project view, select the source table in the Relational Data tab.
2. In the Edit sub-tab, click the Query field.
3. In the Query dialog, specify a SQL statement that limits the amount of data that you load.
For example, to copy one year's worth of data on a table that includes a 
LAST_UPD date column, you might specify (in Oracle Syntax):

SELECT * FROM TABLE_NAME WHERE LAST_UPD > SYSDATE - 365

4. Click OK.

When you provide a SQL query override, Data Sync validates the SQL against the 
database, and prompts you to correct any errors. If the SQL override includes new 
columns that are not present in the table definition, you're prompted to add them to the 
target table.

For example, take a case where a CONTACT table is imported. By default, Data Sync 
issues SELECT * FROM CONTACT. You might want to add a column named 
UPLOADED_DT to the table to record when the data is uploaded. To do this, provide 
a SQL query override such as the following:

SELECT CONTACT.*, SYSDATE AS UPLOADED_DT FROM CONTACT

In this case, Data Sync recognizes that the new column UPLOADED_DT does not 
exist on the target and offers to add it to the table definition.

Loading Data from Tables Using a SQL Query

In Data Sync, you can load data from tables using a SQL query.

You can load data based on a SQL statement. For example, instead of loading detail 
data, you might use an aggregate SQL statement to store compressed data in the 
cloud. This aggregate SQL statement might join multiple tables and use SQL 
functions, such as GROUP BY, filters, and joins.

1. In Data Sync, in the Project view, click the Relational Data tab.
2. Click Data From SQL.
3. In the New Query dialog, enter a logical name for the query in the Name field. 
The name must not contain spaces.
4. Specify an existing target table or create a new one and provide a name for the 
table. 
   If the query defines a new table, the column definitions are inferred from the SQL 
structure. If you use an existing table, any new columns from the SQL can be 
added to the list of columns.
5. Select a connection in the Connection list.
6. Enter the SQL query in the Query window.
7. Click OK.

If you chose to load data incrementally, then a unique index is suggested on the user/ 
primary key columns. It is also recommended that you register additional indexes that 
users can use to join tables and filter reports.

Loading Relational Tables Using SQL Developer

You can use Oracle SQL Developer Release 3.2 or later to upload and administer data 
in Oracle Database Cloud Service.
Oracle SQL Developer is an integrated, transparent, and seamless bulk-data loading facility with full object browsing capabilities. This Java-based tool runs on a client machine and accesses your Oracle Database Cloud Service through a set of RESTful Web Service calls. RESTful Web Service calls enable you to access and load data and data structures.

You can:

- Connect to the database through Oracle SQL Developer.

  **Note:**
  
  To configure Oracle SQL Developer connections for uploading data, refer to the [Data Loading and the Oracle Database Cloud Service tutorial](#).

- Add any Oracle SQL Developer object.
- Move data from on-premises databases to any target Oracle Cloud database.
- Create deployment shopping carts. Oracle SQL Developer creates a cart containing objects that you want to load into your service, connects to your service, and then securely deploys data from the cart to the service. You can also compare the environments and carts.
- Load data incrementally on a schedule by using Oracle SQL Developer.

  **Note:**
  
  Oracle SQL Developer uses the database utility SQL*Loader to perform the data load.

As a guideline, use Oracle SQL Developer to input files greater than 500,000 rows.
Note:

Before loading data into an existing schema or table, consider backing up your data for safekeeping. See Using Oracle Database Backup Cloud Service.
Setting Up Data Loads From OTBI Using Data Sync

Using Data Sync, you can load data directly from subject areas or reports in Oracle Transactional Business Intelligence (OTBI). This enables your users to analyze OTBI data.

Topics

• Typical Workflow for Loading Data from OTBI
• About Loading Data from OTBI Data Sources
• Specifying Connection Details for OTBI Data
• Setting Up Data Loads from OTBI Folders or Subject Areas
• Setting Up Data Loads from Folders Within OTBI Subject Areas
• Setting Up Data Loads from OTBI Using Day-based Partitions

Typical Workflow for Loading Data from OTBI

Here are the common tasks for loading data from OTBI.

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>More Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Get Data Sync up and running</td>
<td>Download and install Data Sync on a local machine, and set up the Data Sync environment.</td>
<td>Setting Up Data Sync for the First Time</td>
</tr>
<tr>
<td>Specify connection details for the data source</td>
<td>Create a connection in Data Sync.</td>
<td>Specifying Connection Details for OTBI Data</td>
</tr>
<tr>
<td>Set up a data load from a report or a subject area</td>
<td>Specify information about your data, such as the format, which columns to load, and how to handle incremental data.</td>
<td>Setting Up Data Loads from OTBI Folders or Subject Areas</td>
</tr>
<tr>
<td>Set up a data load from a folder in a subject area</td>
<td>Specify information about your data, such as the format, which columns to load, and how to handle incremental data.</td>
<td>Setting Up Data Loads from Folders Within OTBI Subject Areas</td>
</tr>
<tr>
<td>Set up a data load from a day-based partition</td>
<td>Specify information about your data, such as the format, which columns to load, partition details, and how to handle incremental data.</td>
<td>Setting Up Data Loads from OTBI Using Day-based Partitions</td>
</tr>
<tr>
<td>Load data using Data Sync</td>
<td>Use Data Sync to load data into your target Cloud database.</td>
<td>Loading Data Using Data Sync</td>
</tr>
<tr>
<td>Refresh your data regularly</td>
<td>Schedule a regular data load to refresh your data.</td>
<td>Refreshing Data Regularly</td>
</tr>
<tr>
<td>Monitor Data Loads</td>
<td>Monitor the progress of data loads and respond to issues.</td>
<td>Monitoring Data Loads</td>
</tr>
</tbody>
</table>
About Loading Data from OTBI Data Sources

You can use Data Sync to load data from OTBI data sources.

What OTBI sources does Data Sync support?

- Oracle Financials Cloud
- Oracle HCM Cloud
- Oracle Procurement Cloud
- Oracle Project Management Cloud
- Oracle Sales Cloud
- Oracle Supply Chain Management Cloud

How do I connect Data Sync to my OTBI data source?

Create a project in Data Sync, and then use the Connections-Sources/Targets dialog to create a connection. See Specifying Connection Details for OTBI Data.

Specifying Connection Details for OTBI Data

To set up a Data Sync environment with an OTBI data source, you specify connection details for your OTBI instance.

1. In Data Sync, click Connections, then click the Sources/Targets tab.
2. Click New to create a new blank row in the table.
3. In the Edit dialog, specify the following details:

<table>
<thead>
<tr>
<th>Field or Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Specify a short descriptive and environment-agnostic name such as SALES_CLOUD to identify the connection in Data Sync.</td>
</tr>
<tr>
<td>Connection Type</td>
<td>Select Oracle BI Connector.</td>
</tr>
<tr>
<td>User</td>
<td>Specify an OTBI user with sufficient administration privileges on the reporting area that you want to load.</td>
</tr>
<tr>
<td>Password</td>
<td>Specify the password for the OTBI user.</td>
</tr>
</tbody>
</table>

4. Click Test Connection, then save your details.

Setting Up Data Loads from OTBI Folders or Subject Areas

Before you start loading data, you specify information about your data, such as the format, which columns to load, and how to handle incremental data.

1. In Data Sync, click the Project tab.
2. Click the Pluggable Source Data tab.
3. Click **Manual Entry** to display the Manual Entry dialog, and specify the following details.

<table>
<thead>
<tr>
<th>Field or Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logical Name</td>
<td>Specify a short description name to identify this data in Data Sync. For example, to load data from a report named Activity Report, you might specify ActivityReportEMEA.</td>
</tr>
<tr>
<td>Target Name</td>
<td>Enter the name that you want to use for the target table. For example, to load data from a report named Activity Report, you might specify OTBI_ACTIVITY_REPORT_EMEA.</td>
</tr>
<tr>
<td>Output Option</td>
<td>Select <strong>Relational</strong> for analysis in enterprise dashboards and analyses. Select <strong>Data Set</strong> for analysis in Data Visualization.</td>
</tr>
<tr>
<td>DB Connection</td>
<td>Select your OTBI data source. This list shows data sources that you specified on the Connections dialog.</td>
</tr>
</tbody>
</table>

4. Click **OK** to display the Message dialog.

5. At the **Data from** option, select the type of pluggable source that you want to load from.
   - To load using a SQL command, select **Logical SQL**.
   - To load from an OTBI report, select **Report**.
   - To load from a table in OTBI, select **Subject Area.Table**.

6. Click **OK** to display the Properties dialog.
   The Properties dialog shows a Name and Value pair for the Report, Subject Area, or SQL statement, depending on the pluggable source type you selected in Step 5.

7. Click the **Value** field to display the Value dialog.

8. Use the Value dialog to specify one of the following:
   - To load using a SQL command, enter the SQL statement. For example,
     
     ```sql
     SELECT    "CRM - Sales Predictor Input"."Customer"."City" s_1,    "CRM - Sales Predictor Input"."Order Item Revenue Facts"."Order Date" s_2,    "CRM - Sales Predictor Input"."Order Item Revenue Facts"."Product Name" s_3 FROM "CRM - Sales Predictor Input"
     ```
   - To load from an OTBI report, enter the full path to the report. For example, `/shared/Custom/Customer Relationship Management/Activity Report`.

   The example screenshot shows the Value configured for an OTBI report.
• To load from a table in OTBI, enter "<Subject Area name> .<Table name> . For example, "CRM - Sales Predictor Input" . Revenue.

9. Click OK.

10. Specify a load strategy:
   a. In the lower pane, click the Edit tab and click the Load Strategy field to display the Load Strategy dialog.
   b. Use the Load Strategy dialog to specify how to load data according to your business needs.

11. To specify how to handle incremental updates, on the lower pane click Mapping, and use the Mapping dialog to configure the data upload according to your business needs.

   If required, you can use the Pluggable Attributes tab to review and update the Name and Value pair for the Report, Subject Area, or SQL statement, depending on the pluggable source type you selected in Step 5.

   You’re now ready to start loading data using a job. Display the Jobs tab and select the job that Data Sync created for your project. Alternatively, create your own job.

Setting Up Data Loads from Folders Within OTBI Subject Areas

Before you start loading data, you specify information about your data, such as the format, which columns to load, and how to handle incremental data.

1. In Data Sync, click the Project tab.

2. Click the Pluggable Source Data tab.

3. Click Data From Object(s) to display the Import Definition into Products dialog, and specify the following details.

<table>
<thead>
<tr>
<th>Field or Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source</td>
<td>Select the OTBI data source where the subject area is located.</td>
</tr>
<tr>
<td>Filter</td>
<td>Use this box to enter a search string using the asterisk character (<em>) as a wildcard. For example, enter Sales</em> to search for all folders with names that start with Sales.</td>
</tr>
<tr>
<td>Output Option</td>
<td>Select Relational for analysis in enterprise dashboards and analyses. Select Data Set for analysis in Data Visualization.</td>
</tr>
</tbody>
</table>

4. Click Search to list all subject areas that match your search string.

5. When the search is complete, select the folders to load:
   • To select individual folders to load, select the Import option next to each folder.
   • To select all folders to load, click Select All.

6. Click Import to load the metadata for the selected folders.

   When the import is complete, a success message is displayed.
7. To view details of imported folders, click the **Target Tables/Data Sets** tab.
   
   In the list of tables and data sets, look in the name column for the folder or folders that you selected for loading.

8. To drill into this target table or data set, click the **Table Columns** tab in the lower pane.

9. Specify a load strategy:
   
   a. In the lower pane, click the **Edit** tab and click the **Load Strategy** field to display the Load Strategy dialog.
   
   b. Use the Load Strategy dialog to specify how to load data according to your business needs.

You’re now ready to start loading data using a job. Display the Jobs tab and select the job that Data Sync created for your project. Alternatively, create your own job.

**Setting Up Data Loads from OTBI Using Day-based Partitions**

You can use Data Sync to load large volumes of OTBI data by chunking the data to make the data load more efficient.

If you’re loading large data volumes, instead of loading all data at once, you can improve load-performance by partitioning the data based on the number of days. When you configure the properties for your data load, if you select the load type ‘Day Based Partitioned Read from Subject Area.Table’ or ‘Day Based Partitioned Read from SQL’, then specify partition details using the Properties dialog.

1. In Data Sync, click the **Project** tab.

2. Click the **Pluggable Source Data** tab.
3. Click **Manual Entry** to display the Manual Entry dialog, and specify the following details.

<table>
<thead>
<tr>
<th>Field or Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logical Name</td>
<td>Specify a short description name to identify this data in Data Sync.</td>
</tr>
<tr>
<td>Target Name</td>
<td>Enter the name that you want to use for the target table.</td>
</tr>
<tr>
<td>Output Option</td>
<td>Select <strong>Relational</strong> for analysis in enterprise dashboards and analyses. Select <strong>Data Set</strong> for analysis in Data Visualization.</td>
</tr>
<tr>
<td>DB Connection</td>
<td>Select your OTBI data source. This list shows data sources that you specified on the Connections dialog.</td>
</tr>
</tbody>
</table>

4. Click **OK** to display the Message dialog.

5. At the **Data from** option, select the partition type (for example, Day Based Partitioned Read from Subject Area.Table).

6. Click **OK** to display the Properties dialog.

   The Properties dialog shows a Name and Value pair for each piece of information that you must specify to define your data partition.

7. Use the **Value** fields to specify the partition details.

   For Day Based Partitioned Read from Subject Area.Table:

<table>
<thead>
<tr>
<th>Property</th>
<th>What to specify</th>
</tr>
</thead>
<tbody>
<tr>
<td>Subject Area.Table</td>
<td>The subject area and table that you want to load.</td>
</tr>
<tr>
<td>Filter</td>
<td>Optionally specify a filter.</td>
</tr>
<tr>
<td>Periodicity Column</td>
<td>A timestamp/date based attribute to partition the reads on. This attribute must not be changed once a record is created.</td>
</tr>
<tr>
<td>Partition Read (Number Of Days)</td>
<td>Specify how many days worth of data should be read at once.</td>
</tr>
</tbody>
</table>

   For Day Based Partitioned Read from SQL

<table>
<thead>
<tr>
<th>Property</th>
<th>What to specify</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial SQL</td>
<td>The SQL to be used when an object is initially extracted</td>
</tr>
<tr>
<td>Incremental SQL</td>
<td>The SQL to be used when an object is incrementally extracted.</td>
</tr>
<tr>
<td>Query to find minimum date</td>
<td>A query to find the minimum date for extracting in partitions. This value will be used as the lower boundary for initial load. For incremental loads, the last refresh date will be used as the lower boundary.</td>
</tr>
<tr>
<td>Query to find maximum date</td>
<td>Find the maximum date for extracting in partitions. This value will be used as the upper boundary for both initial and incremental load.</td>
</tr>
<tr>
<td>Periodicity Column</td>
<td>A timestamp/date based attribute to partition the reads on. This attribute must not be changed once a record is created.</td>
</tr>
<tr>
<td>Partition Read (Number Of Days)</td>
<td>Specify how many days worth of data should be read at once.</td>
</tr>
</tbody>
</table>

8. Click **OK**.
9. Specify a load strategy:
   a. In the lower pane, click the **Edit** tab and click the **Load Strategy** field to display the Load Strategy dialog.
   b. Use the Load Strategy dialog to specify how to load data according to your business needs.

10. To specify how to handle incremental updates, on the lower pane click **Mapping**, and use the Mapping dialog to configure the data upload according to your business needs.

   If required, you can use the Pluggable Attributes tab to review and update the Name and Value pair for the Report, Subject Area, or SQL statement, depending on the pluggable source type you selected in Step 5.

You’re now ready to start loading data using a job. Display the Jobs tab and select the job that Data Sync created for your project. Alternatively, create your own job.
Setting Up Data Loads from JDBC Data Sources Using Data Sync

Use Data Sync to load data from most data sources that support JDBC.

Topics:
- Typical Workflow for Loading Data from JDBC Data Sources
- About Loading Data from JDBC Data Sources
- Specifying Connection Details for Generic JDBC Sources
- Setting Up a Data Load from a JDBC Data Source Using Metadata Objects
- Setting Up a Data Load from a JDBC Data Source Using a Query
- Specifying Connection Details for NetSuite Data

Typical Workflow for Loading Data from JDBC Data Sources

Here are the common tasks for loading data from data sources that support JDBC, such as Greenplum, Salesforce, and Redshift.

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>More Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Get Data Sync up and running</td>
<td>Download and install Data Sync on a local machine, and set up the Data Sync environment.</td>
<td>Setting Up Data Sync for the First Time</td>
</tr>
<tr>
<td>Specify connection details for the data source</td>
<td>Create a connection in Data Sync.</td>
<td>Specifying Connection Details for Generic JDBC Sources</td>
</tr>
<tr>
<td>Set up a data load from a JDBC data source</td>
<td>Specify information about your data, such as the format, which columns to load, and how to handle incremental data. You can load data by object or by query.</td>
<td>Setting Up a Data Load from a JDBC Data Source Using Metadata Objects Setting Up a Data Load from a JDBC Data Source Using a Query</td>
</tr>
<tr>
<td>Load the data using Data Sync</td>
<td>Use Data Sync to load data into your target cloud database.</td>
<td>Loading Data Using Data Sync</td>
</tr>
<tr>
<td>Refresh your data regularly</td>
<td>Schedule regular data loads to refresh your data.</td>
<td>Refreshing Data Regularly</td>
</tr>
<tr>
<td>Monitor data loads</td>
<td>Monitor the progress of data loads and respond to issues.</td>
<td>Monitoring Data Loads</td>
</tr>
</tbody>
</table>
About Loading Data from JDBC Data Sources

You can use Data Sync to load data from most generic JDBC data source types.

What JDBC sources does Data Sync support?

Data Sync is pre-installed with JDBC libraries for commonly used data sources, but you can also install your own JDBC libraries. The pre-installed JDBC libraries are:

- Greenplum
- Hive
- Impala
- Informix
- MongoDB
- PostgreSQL
- Redshift
- Salesforce
- Sybase

These JDBC libraries are installed on the Data Sync client machine in the folder `<Data Sync installation folder>\lib\genericjdbc`. For example, the MongoDB library is `wlmongodb.jar`. If you want to use a different JDBC library, you can install your own JDBC library files in the folder `<Data Sync installation folder>\lib`.

How do I connect Data Sync to my JDBC data source?

Create a project in Data Sync, and then use the Sources/Targets dialog in the Connections view to create a connection. See Specifying Connection Details for Generic JDBC Sources.

How do I specify what data to load?

When you have set up and tested a connection (using the Test Connection option), click the Project tab, then click the Pluggable Source Data tab. You can either select the columns to load using the Data From Objects dialog or specify a query using the Manual Entry dialog.

Can I perform incremental uploads from JDBC data sources?

Yes. To perform incremental extracts, you simply need to include a timestamp attribute, which tracks when a record is inserted or updated.

Supported Data Loading Methods

Data Sync supports four main data loading methods:
<table>
<thead>
<tr>
<th>Data loading method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Query objects in the metadata dictionary (using the <a href="#">Data From Objects</a> option in Data Sync)</td>
<td>If your JDBC driver supports the querying of the metadata dictionary, then you can use this method to select from available objects. This is similar to the <a href="#">Data From Tables</a> option on the Relational Data tab. You can either query the whole metadata dictionary, or if you know the list of objects, you can also selectively import specific objects by selecting <strong>Type list of object names</strong> and specifying the objects. In addition you can define an optional filter condition that limits the rows from the object. The filter clause should specify the condition only. For example, to extract contacts who live in CA, specify &quot;STATE = 'CA'&quot;. Do not include the &quot;WHERE&quot; keyword. The filter clause can be a complex or nested expression that can be processed by the supporting data source.</td>
</tr>
<tr>
<td>Specify a query (using the Manual Entry dialog in Data Sync and selecting <strong>Query</strong>)</td>
<td>Specify a query whose results can be replicated to a table. You must specify a query with a WHERE clause only, as additional filters can be appended to the query. Do not use a query with 'order by' or 'having' or 'group by' clauses. If necessary, use a query with sub-queries. For example, select contact_name, order_dt from (select contact_name, max(order_dt) from orders group by contact_name) recent_orders.</td>
</tr>
<tr>
<td>Specify a partition based on objects (using the Manual Entry dialog in Data Sync and selecting <a href="#">Day-based partitioned read from an object</a>)</td>
<td>Use this method if you cannot load all of the data at once. Specify the records to be read for a specified number of days at a time based on a timestamp column. Specify: – Object Name - the name of the table or object in the data source. – Periodicity Column - a column/Attribute whose data type is either date/timestamp which can be used to partition the data. – Partition Read (Number of Days) - the number of days of data to read at a time. – Filter Conditions - filter conditions to limit the data. Do not use the &quot;WHERE&quot; key word. The filter clause can be a complex or nested expression that can be processed by the supporting data source.</td>
</tr>
<tr>
<td>Specify a partition based on a SQL query (using the Manual Entry dialog in Data Sync and selecting <a href="#">Day-based partitioned read from an query</a>)</td>
<td>Use this method if you can’t load all of the data at once. You can specify the records to be read for a specified number of days at a time based on a timestamp column. Specify: – Object Name - the name of the table or object in the data source. – Periodicity Column - a column or attribute whose data type is either date or timestamp that can be used to partition the data. – Partition Read (Number of Days) - the number of days of data to read at a time. – Filter Conditions - filter conditions to limit the data. The filter clause can be a complex or nested expression that can be processed by the supporting data source. Do not use the &quot;WHERE&quot; key word.</td>
</tr>
</tbody>
</table>
Specifying Connection Details for Generic JDBC Sources

With Data Sync you can load data from many commonly used JDBC sources.

What information do I need to connect Data Sync to my JDBC data source?

<table>
<thead>
<tr>
<th>Field</th>
<th>What to specify</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit tab — Name</td>
<td>A short string to identify this connection in Data Sync.</td>
</tr>
<tr>
<td>Edit tab — Connection Type</td>
<td>Generic JDBC</td>
</tr>
<tr>
<td>Edit tab — User and Password</td>
<td>User and password for access to the data source. Make sure that the user has sufficient administration privileges on the reporting area that you want to load.</td>
</tr>
<tr>
<td>Edit tab — URL</td>
<td>Specify the URL for your JDBC data source using the information from the URL column in the Example Drivers and URLs table. For example, for GreenPlum, a URL in this format: jdbc:oracle:.greenplum://hostname:[port]</td>
</tr>
<tr>
<td>Edit tab — JDBC Driver</td>
<td>Copy the appropriate driver information from the Driver column in the Example Drivers and URLs table. For example, for GreenPlum, enter com.oracle.bi.jdbc.greenplum.GreenplumDriver</td>
</tr>
<tr>
<td>Advanced Properties tab — Enclose object names</td>
<td>If the object names in your data source (for example, tables, column, indexes) contain spaces or special characters, then specify the opening and closing character separated by a comma. For example, if your names are enclosed in square brackets ([ and ]), specify: [,]</td>
</tr>
</tbody>
</table>

How do I specify what data to load?

When you have set up and tested the connection (using the Test Connection option), follow the steps in About Loading Data from JDBC Data Sources.

Example Drivers and URLs

<table>
<thead>
<tr>
<th>Source</th>
<th>Driver</th>
<th>URL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hive</td>
<td>com.oracle.bi.jdbc.hive.HiveDriver</td>
<td>jdbc:oracle:hive://HOST_NAME:PORT_NUMBER;DatabaseName=DATABASE_NAME</td>
</tr>
<tr>
<td>MongoDB</td>
<td>com.oracle.bi.jdbc.mongodb.MongoDBDriver</td>
<td>jdbc:oracle:mongodb://HOST_NAME:PORT_NUMBER;DatabaseName=DATABASE_NAME</td>
</tr>
<tr>
<td>Source</td>
<td>Driver</td>
<td>URL</td>
</tr>
<tr>
<td>----------</td>
<td>------------------------------------------------------------------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>NetSuite</td>
<td>com.netsuite.jdbc.openaccess.OpenAccessDriver</td>
<td>jdbc:ns://{Server Host}:{Server Port};ServerDataSource={Server Data Source};encrypted=1;Ciphersuites={Cipher Suite};CustomProperties=(AccountID={Account Id};RoleID={Role Id})</td>
</tr>
<tr>
<td>Postgres</td>
<td>com.oracle.bi.jdbc.postgresql.PostgreSQLDriver</td>
<td>jdbc:oracle:postgresql://HOST_NAME:PORT_NUMBER;DatabaseName=DATABASE_NAME</td>
</tr>
<tr>
<td>Redshift</td>
<td>com.oracle.bi.jdbc.redshift.RedshiftDriver</td>
<td>jdbc:oracle:redshift://REDSHIFT_ENDPOINT:PORT_NUMBER;DatabaseName=dev</td>
</tr>
<tr>
<td>Salesforce</td>
<td>com.oracle.bi.jdbc.sforce.SForceDriver</td>
<td>jdbc:oracle:sforce://&lt;ServerName&gt;;SecurityToken=&lt;Security token&gt;</td>
</tr>
<tr>
<td>Sybase</td>
<td>com.oracle.bi.jdbc.sybase.SybaseDriver</td>
<td>jdbc:oracle:sybase://HOST_NAME:PORT_NUMBER;DatabaseName=DATABASE_NAME</td>
</tr>
</tbody>
</table>

Additional Information

- When you specify a Salesforce URL, you need the security token for the Salesforce user account being used, which was emailed to the user when the account was set up.
- When you specify connection details for a Sybase data source, sometimes you also have to specify a schema owner or table owner (using the Schema/Table Owner field).

Setting Up a Data Load from a JDBC Data Source Using Metadata Objects

In Data Sync, you can specify the columns that you want to load, and optionally specify a data filter that selects a specific sub-set of data.

1. In Data Sync, click Project, display the Pluggable Data Source tab, then click Data from Object(s).
2. Click Discover objects by listing.
3. At the Import Definition dialog, select the JDBC connection that you created for your data source in the Source list, and use the Filter field to specify the first few characters of the column that you want to load, plus the wildcard character (*), then click Search. For example, to search for incidents, you might enter incid*.
4. Select the objects that you want to load by clicking the Import Definition check box for each object, select the appropriate Output Option, then click Import.
5. When the import is complete, select the new record in the Pluggable Source Data list.
6. Display the Pluggable Attributes sub-tab, and specify the attributes.
Setting Up a Data Load from a JDBC Data Source Using a Query

In Data Sync, you can specify the columns that you want to load, and optionally specify a data filter that selects a specific sub-set of data.

1. In Data Sync, click Project, then display the Pluggable Data Source tab.
2. Click Manual Entry, and specify the report details.

3. At the Message dialog, select the type of query to use from the Data from list.

A Properties dialog presents a list of Name and Value pairs for the query type selected. For example, if you select Query, the properties dialog displays two Name and Value pairs.

Similarly, if you select Day-based partitioned read from an object, the properties dialog displays five Name and Value pairs.
4. For each Name and Value pair displayed on the properties dialog (except READ_TYPE, which is read-only), click the Value field and enter a value.

For example, if you select Day-based partitioned read from an object from the previous dialog, you define a value for Object Name, Periodicity Column, Partition Read (Number of days), and Filter Condition(s).

5. Review the new data source on the Pluggable Source Data page

Use the Pluggable Attributes tab to verify the query details.

6. Specify a load strategy:
   a. In the lower pane, click the Edit tab and click the Load Strategy field to display the Load Strategy dialog.
   b. Use the Load Strategy dialog to specify how to load data according to your business needs.

You're now ready to start loading data using a job. Display the Jobs tab and select the job that Data Sync created for your project. Alternatively, create your own job.

Specifying Connection Details for NetSuite Data

To set up a Data Sync environment with a NetSuite data source, you specify connection details for your NetSuite instance.

1. Install the NetSuite JDBC driver:
   a. Download the JDBC driver installer from NetSuite and install it.
   b. From the installed directory, copy NQjc.jar to the \lib folder in your Data Sync installation directory.
   c. If Data Sync is already running, exit and restart.

2. In Data Sync, click Connections.
3. Click New to create a new blank row in the table.
4. On the Edit tab, specify the following details:
<table>
<thead>
<tr>
<th>Field or Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Specify a short descriptive name such as NetSuite to identify the connection details in Data Sync.</td>
</tr>
<tr>
<td>Connection Type</td>
<td>Select <strong>Generic JDBC</strong>.</td>
</tr>
<tr>
<td>User</td>
<td>Specify a NetSuite user with sufficient administration privileges on the reporting area that you want to load.</td>
</tr>
<tr>
<td>Password</td>
<td>Specify the password for the NetSuite user.</td>
</tr>
</tbody>
</table>
| URL             | Specify:  
|                 | jdbc:ns://{Server Host}:{Server Port};ServerDataSource={Server Data Source};encrypted=1;Ciphersuites={Cipher Suite};CustomProperties={AccountID={Account Id};RoleID={Role Id}} |

For example:

jdbc:ns://my.netsuite.com:1708;ServerDataSource=NetSuite.com;encrypted=1;Ciphersuites=TLS_RSA_WITH_AES_128_CBC_SHA;CustomProperties={AccountID=TSTDRV1660232;RoleID=3}

5. **Click Test Connection**, then save your details.
Setting Up Data Loads From Oracle Service Cloud (RightNow)

Use Data Sync to load data from Oracle Service Cloud (RightNow). This enables your users to analyze RightNow data.

Topics:
- Typical Workflow for Loading Data from Oracle Service Cloud (RightNow)
- About Loading Data From Oracle Service Cloud (RightNow)
- Specifying Connection Details for Oracle Service Cloud (RightNow)
- Setting Up A Data Load From An Oracle Service Cloud (RightNow) Report
- Setting Up a Data Load From Oracle Service Cloud (RightNow) Using a ROQL Query
- Setting Up a Data Load From Oracle Service Cloud (RightNow) Using a Metadata Query
- Setting Up a Data Load From Oracle Service Cloud (RightNow) Using Named IDs

Typical Workflow for Loading Data from Oracle Service Cloud (RightNow)

Here are the common tasks for loading data from Oracle Service Cloud (RightNow).

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>More Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Get Data Sync up and running</td>
<td>Download and install Data Sync on a local machine, and set up the Data Sync environment.</td>
<td>Setting Up Data Sync for the First Time</td>
</tr>
<tr>
<td>Specify connection details for the data source</td>
<td>Create a connection in Data Sync.</td>
<td>Specifying Connection Details for Oracle Service Cloud (RightNow)</td>
</tr>
<tr>
<td>Decide which data load configuration to use</td>
<td>Choose one of the data loading configurations that Data Sync supports.</td>
<td>About Loading Data From Oracle Service Cloud (RightNow)</td>
</tr>
<tr>
<td></td>
<td>• Using a Report</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Using a ROQL Query</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Using Objects</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Using Named Field IDs</td>
<td></td>
</tr>
<tr>
<td>Task</td>
<td>Description</td>
<td>More Information</td>
</tr>
<tr>
<td>---------------------------</td>
<td>------------------------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Set up your data load</td>
<td>Specify information about your data, such as the format, which columns to load, and how to handle incremental data.</td>
<td>Using a Report – Setting Up A Data Load From An Oracle Service Cloud (RightNow) Report</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Using a ROQL Query – Setting Up a Data Load From Oracle Service Cloud (RightNow) Using a ROQL Query</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Using Objects – Setting Up a Data Load From Oracle Service Cloud (RightNow) Using a Metadata Query</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Using Named Field IDs – Setting Up a Data Load From Oracle Service Cloud (RightNow) Using Named IDs</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Load the data using Data Sync</td>
<td>Use Data Sync to load data into your target cloud database.</td>
<td>Loading Data Using Data Sync</td>
</tr>
<tr>
<td>Refresh your data regularly</td>
<td>Schedule regular data loads to refresh your data.</td>
<td>Refreshing Data Regularly</td>
</tr>
<tr>
<td>Monitor data loads</td>
<td>Monitor the progress of data loads and respond to issues.</td>
<td>Monitoring Data Loads</td>
</tr>
</tbody>
</table>

### About Loading Data From Oracle Service Cloud (RightNow)

You can use Data Sync to load data from Oracle Service Cloud (RightNow).

**What are the different options for loading data and how do I choose the best option?**

<table>
<thead>
<tr>
<th>Data Load Method</th>
<th>When to use</th>
</tr>
</thead>
<tbody>
<tr>
<td>Using Reports</td>
<td>If you’re familiar with the RightNow desktop application, then this method is relatively easy to set up. See Setting Up A Data Load From An Oracle Service Cloud (RightNow) Report.</td>
</tr>
<tr>
<td>Using ROQL</td>
<td>If you know precisely what attributes you want to replicate, and are familiar with writing ROQL statements. See Setting Up a Data Load From Oracle Service Cloud (RightNow) Using a ROQL Query.</td>
</tr>
<tr>
<td>Using Objects</td>
<td>If you know the object that you would like to replicate. Optionally you can specify filters to load a subset of the rows. See Setting Up a Data Load From Oracle Service Cloud (RightNow) Using a Metadata Query.</td>
</tr>
<tr>
<td>Using Named IDs</td>
<td>If you only want to load specific fields and you know the field IDs. See Setting Up a Data Load From Oracle Service Cloud (RightNow) Using Named IDs.</td>
</tr>
</tbody>
</table>
Which version of RightNow do I need?

You can upload data from RightNow Version 15.11 or later. You'll have to upgrade, if you have an older version. You load data from the reporting instance of RightNow, not the transactional instance.

How do I connect Data Sync to my RightNow data source?

Create a project in Data Sync, and then use the Connections-Sources/Targets dialog to create a connection. See Specifying Connection Details for Oracle Service Cloud (RightNow).

How do I generate a data report definition from my RightNow data source?

- Decide what you data you need and how much data you need to analyze. For example, you might have five years of data in your RightNow application but you might want to analyze data for the most recent year only.
- Use the RightNow desktop application to create a data report definition.
- Include timestamp data for incremental refresh, and a numeric ID that is used to load the data in manageable chunks.
- In addition to the data report, create a metadata report that includes attributes named MAX_VALUE, MIN_VALUE, and COUNT.
- Keep a note of the unique report IDs. You'll need to specify these when you set up data loads in Data Sync. See Setting Up A Data Load From An Oracle Service Cloud (RightNow) Report.

Can I perform incremental uploads from RightNow?

Yes. To perform incremental extracts, you simply need to include a field such as date created or last updated date from the table that you're referencing for the report creation.

Filtering your RightNow data on timestamps

Whichever way you load your RightNow data (by report, ROQL query, or object), you use a filter to specify the data that you want your users to analyze. For example, you might have five years of data in RightNow but you only want to analyze data for the most recent year.

When you specify a timestamp in a query, the timestamp must be in the format:

`yyyy-MM-ddTHH:mm:ssZ`

For example, you might filter a query using:

```
updatedtime > '2014-01-01T00:00:00Z'.
```
Specifying Connection Details for Oracle Service Cloud (RightNow)

With Data Sync you can load data from Oracle Service Cloud (RightNow).

What information do I need to connect Data Sync to my RightNow data source?

<table>
<thead>
<tr>
<th>Field</th>
<th>What to specify</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>A short string to identify the connection in Data Sync.</td>
</tr>
<tr>
<td>Connection Type</td>
<td>Oracle Service Cloud (RightNow)</td>
</tr>
<tr>
<td>User and Password</td>
<td>Database user and password for access to the data source. Make sure that the user has sufficient administration privileges on the reporting area that you want to load.</td>
</tr>
<tr>
<td>URL</td>
<td>Specify the URL for your RightNow instance, for example, <a href="https://integration-test.rightnowdemo.com/">https://integration-test.rightnowdemo.com/</a></td>
</tr>
<tr>
<td>Timezone</td>
<td>UTC00:00 (recommended).</td>
</tr>
</tbody>
</table>

How do I specify what data to load?

When you have set up and tested the connection (using the Test Connection option), follow the steps in About Loading Data From Oracle Service Cloud (RightNow).

Setting Up A Data Load From An Oracle Service Cloud (RightNow) Report

Using Data Sync, you can load data using an Oracle Service Cloud (RightNow) report definition that you created using the RightNow desktop application.

Before you start, generate a report containing the data you want to load and note down the report ID of the data report and the report ID of the associated helper report. See About Loading Data From Oracle Service Cloud (RightNow).

1. In Data Sync, click Project, then display the Pluggable Data Source tab.
2. Click Manual Entry, and specify the report details.

<table>
<thead>
<tr>
<th>Field or Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logical Name</td>
<td>Specify a short meaningful name to identify the data load details in Data Sync. For example, Incidents_from_RightNow_report. The name must not contain spaces, and must be different from the Target Name.</td>
</tr>
<tr>
<td>Target Name</td>
<td>Specify a short meaningful name to identify the data load details in Data Sync. For example, INCIDENTS_REPORT. The name must not contain spaces, and must be different from the Logical Name.</td>
</tr>
<tr>
<td>Output Option</td>
<td>Select Relational for analysis in enterprise dashboards and analyses. Select Data Set for analysis in Data Visualization.</td>
</tr>
</tbody>
</table>
3. Click **OK** select Analytics Reports from the **Data from** list.

4. In the Properties dialog, specify the report details.

<table>
<thead>
<tr>
<th>Field or Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DB Connection</td>
<td>Select the connection that you created to your RightNow instance, for example MyRightNow.</td>
</tr>
<tr>
<td>Analytics Report id</td>
<td>Enter the ID for the data report, for example, 100777.</td>
</tr>
<tr>
<td>Helper Analytics Report id</td>
<td>Enter ID for the associated metadata report (containing MAX_VALUE, MIN_VALUE, and COUNT), for example, 100779.</td>
</tr>
<tr>
<td>Numeric Column</td>
<td>The name of the numeric column that you’re using to load your data in manageable chunks. For example, Incident ID.</td>
</tr>
<tr>
<td>Maximum number of rows to read at a time</td>
<td>Leave the default value.</td>
</tr>
</tbody>
</table>

5. Specify a load strategy:

   a. In the lower pane, click the **Edit** tab and click the **Load Strategy** field to display the Load Strategy dialog.

   b. Use the Load Strategy dialog to specify how to load data according to your business needs.

Review the new data source on the Pluggable Source Data page, and look at the Pluggable Attributes tab to verify the report details.

You're now ready to start loading data using a job. Display the Jobs tab and select the job that Data Sync created for your project. Alternatively, create your own job.

### Setting Up a Data Load From Oracle Service Cloud (RightNow) Using a ROQL Query

Using Data Sync, you can load data from Oracle Service Cloud (RightNow) using a ROQL (RightNow Object Query Language) query.

1. In Data Sync, click **Project**, then display the **Pluggable Data Source** tab.

2. Click **Manual Entry**, and specify the details.

<table>
<thead>
<tr>
<th>Field or Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logical Name</td>
<td>Specify a short meaningful name to identify the data load details in Data Sync. For example, Incidents_from_RightNow_ROQL. The name must not contain spaces, and must be different from the <strong>Target Name</strong>.</td>
</tr>
<tr>
<td>Target Name</td>
<td>Specify a short meaningful name to identify the data load details in Data Sync. For example, INCIDENTS_ROQL. The name must not contain spaces, and must be different from the <strong>Logical Name</strong>.</td>
</tr>
</tbody>
</table>
3. Click **OK** and select **ROQL** from the **Data from** list.

4. In the Properties dialog, specify the report details.

<table>
<thead>
<tr>
<th>Field or Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output Option</td>
<td>Select <strong>Relational</strong> for analysis in enterprise dashboards and analyses. Select <strong>Data Set</strong> for analysis in Data Visualization.</td>
</tr>
<tr>
<td>DB Connection</td>
<td>Select the connection that you created for your RightNow instance, for example MyRightNow.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Field or Element</th>
<th>Description</th>
</tr>
</thead>
</table>
| **ROQL Tabular Query**            | Specify a ROQL SQL statement that defines the data you want to load. Use a WHERE clause to filter the data that you want to analyze. For example, to load data from the most recent year, you might enter: `SELECT * FROM incidents WHERE updatedtime > '2014-01-01T00:00:00Z'`  
**Note:** Do not include a GROUP BY clause or similar aggregate function at the end of the SQL statement because this will prevent Data Sync from partitioning the data. |
| **ROQL Tabular Query Objects**    | Specify the RightNow native objects that you want to load, separated by a comma. For example, `location, incidents`.  
For any objects that are used but not specified here, Data Sync defaults the datatype to VARCHAR(200). Use the **Project > Target Tables/Data Sets > Table Columns** tab to verify data types and update the data type if required. |
| **Numeric Column**                | Specify the name of the numeric column that you're using to load data in manageable chunks, for example, `id`. |
| **Get Maximum Numeric Value Query** | Specify a ROQL query to obtain the maximum value for the column specified in the **Numeric Column** field, using the same WHERE clause that you used to limit the data in the **ROQL Tabular Query** field. This query obtains the upper boundary of the data filter. For example, to load data from the most recent year, you might enter: `SELECT MAX(ID) FROM incidents WHERE updatedtime > '2014-01-01T00:00:00Z'`  
**Note:** Do not include a GROUP BY clause or similar aggregate function at the end of the SQL statement because this will prevent Data Sync from partitioning the data. |
| **Get Minimum Numeric Value Query** | Specify a ROQL query to obtain the minimum value for the column specified in the **Numeric Column** field, using the same WHERE clause that you used to limit the data in the **ROQL Tabular Query** field. This query obtains the lower boundary of the data filter. For example, to load data from the most recent year, you might enter: `SELECT MIN(ID) FROM incidents WHERE updatedtime > '2014-01-01T00:00:00Z'`  
**Note:** Do not include a GROUP BY clause or similar aggregate function at the end of the SQL statement because this will prevent Data Sync from partitioning the data. |
### Field or Element | Description
--- | ---
Get Total Count Query | Specify a ROQL query to obtain the number of rows for the column specified in the **Numeric Column** field, using the same WHERE clause that you used to limit the data in the **ROQL Tabular Query** field. This query obtains the number of rows in your filtered data. For example, to load data from the most recent year, you might enter:
```
SELECT COUNT(*) FROM incidents WHERE updatedtime > '2014-01-01T00:00:00Z'
```
**Note:** Do not include a GROUP BY clause or similar aggregate function at the end of the SQL statement because this will prevent Data Sync from partitioning the data.

| Maximum number of rows to read at a time | Leave the default value. |

For example, to specify a query in the **ROQL Tabular Query** field, click the Value field next to the **ROQL Tabular Query** field, and use the Value dialog to enter the query.

5. **Specify a load strategy:**
   a. In the lower pane, click the **Edit** tab and click the **Load Strategy** field to display the Load Strategy dialog.
   b. Use the Load Strategy dialog to specify how to load data according to your business needs.

Review the new data source on the Pluggable Source Data page, and look at the Pluggable Attributes tab to verify the query details.

You’re now ready to start loading data using a job. Display the Jobs tab and select the job that Data Sync created for your project. Alternatively, create your own job.
Setting Up a Data Load From Oracle Service Cloud (RightNow) Using a Metadata Query

Using Data Sync, you can load data from Oracle Service Cloud (RightNow) using a metadata query on the schema objects.

1. In Data Sync, click **Project**, display the **Pluggable Data Source** tab, then click **Data from Object(s)**.
2. Click **Discover objects by listing**, then click **OK**.
3. At the Import Definition dialog, select RightNow in the **Source** list, and use the **Filter** field to specify the first few characters of the RightNow column that you want to load, plus the wildcard character (*), then click **Search**. For example, to search for incidents, you might enter `incid*`.
4. Select the columns that you want to load, click the **Import Definition** check box for each column, then click **Import**.
5. When the import is complete, select the new record in the Pluggable Source Data list.
6. Display the Pluggable Attributes sub-tab, and specify the attributes.

<table>
<thead>
<tr>
<th>Field or Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROQL Query Conditions</td>
<td>Specify a WHERE clause to limit the amount of data returned to a manageable size. For example, to load data from the most recent year, you might enter: <code>updatedtime &gt; '2014-01-01T00:00:00Z'</code>.</td>
</tr>
<tr>
<td>Numeric Column</td>
<td>Specify the name of the numeric column that you’re using to load data in manageable chunks, for example, <code>id</code>.</td>
</tr>
<tr>
<td>Maximum number of rows to read at a time</td>
<td>Don’t change the default value.</td>
</tr>
</tbody>
</table>

7. Specify a load strategy:
   a. In the lower pane, click the **Edit** tab and click the **Load Strategy** field to display the Load Strategy dialog.
   b. Use the Load Strategy dialog to specify how to load data according to your business needs.

Review the new data source on the Pluggable Source Data page, and look at the Pluggable Attributes tab to verify the report details.

You’re now ready to start loading data using a job. Display the Jobs tab and select the job that Data Sync created for your project. Alternatively, create your own job.

Setting Up a Data Load From Oracle Service Cloud (RightNow) Using Named IDs

Using Data Sync, you can load data from Oracle Service Cloud (RightNow) using named field IDs. For example, you might want to load only
incidents.assignedTo.staffGroup and
incidents.banner.importanceFlag.

1. In Data Sync, click **Project**, then display the **Pluggable Data Source** tab.
2. Click **Manual Entry**, and specify the details.

<table>
<thead>
<tr>
<th>Field or Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logical Name</td>
<td>Specify a short meaningful name to identify the data load details in Data Sync. For example, Incidents_from_RightNow_IDs. The name must not contain spaces, and must be different from the Target Name.</td>
</tr>
<tr>
<td>Target Name</td>
<td>Specify a short meaningful name to identify the data load details in Data Sync. For example, INCIDENTS_IDS. The name must not contain spaces, and must be different from the Logical Name.</td>
</tr>
<tr>
<td>Output Option</td>
<td>Select <strong>Relational</strong> for analysis in enterprise dashboards and analyses. Select <strong>Data Set</strong> for analysis in Data Visualization.</td>
</tr>
<tr>
<td>DB Connection</td>
<td>Select the connection that you created for your RightNow instance, for example MyRightNow.</td>
</tr>
</tbody>
</table>

3. Click **OK**, then select **Named Ids** from the **Data from** list.
4. In the Properties dialog, specify the report details.

<table>
<thead>
<tr>
<th>Field or Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Named Id Listing</td>
<td>Click <strong>Value</strong>, and use the Value dialog to specify a one or more field IDs, with each ID on a new line. For example: incidents.assignedTo.staffGroup incidents.banner.importanceFlag</td>
</tr>
<tr>
<td>Insert unspecified row for every Named Id</td>
<td>Specify true.</td>
</tr>
</tbody>
</table>

For example, to specify a query in the **ROQL Tabular Query** field, click the Value field next to the **ROQL Tabular Query** field, and use the Value dialog to enter the query.

5. Specify a load strategy:
   a. In the lower pane, click the **Edit** tab and click the **Load Strategy** field to display the Load Strategy dialog.
   b. Use the Load Strategy dialog to specify how to load data according to your business needs.

Review the new data source on the Pluggable Source Data page, and look at the Pluggable Attributes tab to verify the report details.

You’re now ready to start loading data using a job. Display the Jobs tab and select the job that Data Sync created for your project. Alternatively, create your own job.
Automating Loading Data Using Programmatic Methods

This topic describes how to use programmatic methods to load your data. Generally, you use these methods to load large volumes of data, perform complex transformations, create indexes, and perform database management and monitoring tasks.

Topics:

- About the Oracle BI Cloud Service REST API
- About the Oracle Database Cloud Service API
- About PL/SQL Database Scripts

About the Oracle BI Cloud Service REST API

You use the Oracle BI Cloud Service REST API to define or customize your own API and programmatically load on-premises data for analysis in Oracle BI Cloud Service. The Oracle BI Cloud Service REST API is optimized for loading large volumes of data (thousands to millions) from one or more sources.

You can access the API at REST API for Oracle BI Cloud Service.

Load data to tables on Oracle Database Cloud Service:

The REST API for Oracle BI Cloud Service loads data into the default cloud database connected to Oracle BI Cloud Service. Once data is loaded on to this cloud database, you can:

- Insert, update, upsert, and delete large numbers of records by streaming batches of records.
- Update statistics.
- Drop or create indices on tables on the Oracle Database Cloud Service.
- Write programs and scripts in your favorite programming language and then combine them with additional application logic to invoke the Oracle BI Cloud Service REST API.
- Invoke the script or application with your on-premises scheduler.
- Automate incremental data loading or integrate with ETL tools using the Oracle BI Cloud Service REST API. See REST API for Oracle BI Cloud Service.
- Create a customized API using the Oracle BI Cloud Service REST API framework.

Load data to data sets on Oracle BI Cloud Service:

- Create, update, and delete data sets from on-premises data sources.
- Manage data sets programmatically. Delete unwanted data sets to free up storage space.
About the Oracle Database Cloud Service REST API

If Oracle BI Cloud Service integrates with Oracle Database Cloud - Database Schema Service you can load data programmatically using the Oracle Database Cloud Service REST API.

Create RESTful Web Services to access SQL and PL/SQL queries in Oracle Database Cloud - Database Schema Service from outside the cloud. You use the RESTful Web Services wizard to create RESTful Web Services. The RESTful wizard provides a mechanism to access the service and enables you to define a set of Uniform Resource Identifiers (URIs) to a SQL query or PL/SQL script. You can call out to any SQL query to read any data and return a result, or call out to PL/SQL code to read, write, modify, or delete data.

You can define your own REST API that can be invoked from on-premises environments. You can define any PL/SQL block with BIND variables that can modify tables in Oracle Database Cloud - Database Schema Service.

In addition, you can use the APEX_WEB_SERVICE API within a PL/SQL block to invoke any REST or SOAP API that is supported by cloud applications and retrieve data from an external system. The data can be transferred directly from the on-premises source to the cloud database. You can then programmatically automate the web service.

See Implementing RESTful Web Services in Using Oracle Database Cloud - Database Schema Service.
About PL/SQL Database Scripts

Oracle Database Cloud - Database Schema Service only. You can use PL/SQL database scripts to load data from external, generic web services.

You can retrieve and load data from other cloud services, such as Oracle’s Fusion Cloud or Salesforce.com. The API can be called from your PL/SQL code created by using SQL Workshop, a robust developer’s tool that you use to create user-friendly, front-end applications to execute complex database operations (for example, data entry applications that execute PL/SQL to pass parameters entered by users).

SQL Workshop is part of Oracle Application Express (APEX). Oracle APEX is Oracle’s primary tool for developing web applications with SQL and PL/SQL. Using only a web browser, you can develop and deploy professional Web-based applications for desktops and mobile devices.

You can build PL/SQL scripts in SQL Workspace that:

- Author and execute SQL queries against objects in the Database Schema Service.
- Author and execute database procedures.
- Maintain database objects. For example, you can:
  - Create and maintain indices to improve performance.
  - Drop unneeded objects, such as tables.
  - Create views to expose functionality in the database.
  - Add columns.
  - Modify objects created post-load, such as conform date types.
  - Perform post-load transformations.

See SQL Syntax in Using Oracle Database Cloud - Database Schema Service.
Transforming Your Data

Use Data Sync to transform your relational data as you load it. For example, you can apply default values, calculations, conversions, concatenations, splits, SQL commands, joins and lookups, and create new target data columns based on any supported SQL expressions.

Note:
You can use Data Sync to transform relational data if you're loading data into either Database As A Service or an on-premises database that is configured using the ‘Oracle (Thin)’ connection type.

Tutorial

Topics:
- Typical Workflow for Transforming Data Using Data Sync
- About Transforming Your Data
- Transforming Your Data With Default Values, Conversions, and Calculations
- Transforming Your Data With New Target Columns
- Transforming Your Data Using Joins
- Transforming Your Data Using Surrogate Keys
- Tracking Information About Your Data
- Manipulating Your Data Before And After Data Loads

Typical Workflow for Transforming Data Using Data Sync

Here are the common tasks for transforming data.

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>More Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configure your data load as normal</td>
<td>Configure a data load for your data source type. For example, your source data might be file-based, in relational tables, OTBI, or Oracle Service Cloud.</td>
<td>Refer to the configuration instructions appropriate for your data source type.</td>
</tr>
</tbody>
</table>
### About Transforming Your Data

You can use Data Sync to transform your data.

#### Before You Start

You can use Data Sync to transform relational data if you’re loading data into either Database As A Service or an on-premises database that is configured using the ‘Oracle (Thin)’ connection type. Data Sync always performs transformations on the target database. Data Sync doesn’t support transformations for Oracle BI Cloud Service targets or other target database types.

### Transforming Your Data With Default Values, Conversions, and Calculations

Use Data Sync to transform and cleanse your relational data.

For example, you can apply default values, calculations, conversions, and concatenations.

#### Note:

You can use Data Sync to transform relational data if you’re loading data into either Database As A Service or an on-premises database that is configured using the ‘Oracle (Thin)’ connection type.

1. Create a new Project for your data and configure the data load.
For example, for file-based data, use the File Data tab, or for relational data, use the Relational Data tab.

2. In the Project view, display the Mapping or Column Mapping dialog:
   - If you’re loading file-based data, click **File Targets**, then click **Column Mapping**.
   - If you’re loading from a relational data source or a pluggable data source, then click **Mapping**.

3. Select a column to edit.

4. Apply your transformation:
   - To apply a default value, click **Target Expression** to display the Expression dialog, then click **Default** and enter the value that you want to store in the target column.
     For example, enter 0, or enter No value.
   - To calculate a value, click **Target Expression** to display the Expression dialog, and enter a SQL expression.
     For example, to calculate a Return on Investment (ROI) value, you might enter 
     \[(\text{REVENUE} \times (\text{DISCNT\_RATE}/100)) - \text{COST}\].
   - To convert a value, click **Target Expression** to display the Expression dialog, and enter a SQL expression.
     Examples: To concatenate two columns, you might enter \[\text{TITLE} \ | \ | \text{FIRSTNAME} \ | \ | \text{LASTNAME}\]. To convert a timestamp in ORDER\_DAY\_DT, you might enter \[\text{TO\_NUMBER(TO\_CHAR(ORDER\_DAY\_DT, 'YYYYMMDD'))}\].
     To convert \text{LASTNAME} to upper-case, you might enter \[\text{UPPER(LASTNAME)}\].

### Transforming Your Data With New Target Columns

Use Data Sync to create a new column in your target database.

For example, you might calculate return on investment and store the value in a new column.

**Note:**
You can use Data Sync to transform relational data if you're loading data into either Database As A Service or an on-premises database that is configured using the 'Oracle (Thin)' connection type.

1. Create a new Project for your data and configure the data load.
   For example, for file-based data, use the File Data tab; for relational data, use the Relational Data tab.

2. In the Project view, display the Mapping or Column Mapping dialog:
   - If you're loading file-based data, click **File Targets**, then click **Column Mapping**.
   - If you're loading from a relational data source or a pluggable data source, then click **Mapping**.
3. Click New, and specify the details of the column that you want to create. For example, specify a name, type, target name, and so on.

4. Click Target Expression, and use the Expression dialog to specify a SQL expression that defines your target column. For example, for a return on investment value, you might enter (REVENUE * (DISCNT_RATE/100)) – COST.

5. Click Unmapped Columns, and add the new column to the Selected Columns list.

Transforming Your Data Using Surrogate Keys

Use Data Sync to improve performance by creating surrogate keys.

For example, if your source data contains a variable-length email address, you might create a numeric surrogate key that makes data loading more efficient.

**Note:**

You can use Data Sync to transform relational data if you’re loading data into either Database As A Service or an on-premises database that is configured using the ‘Oracle (Thin)’ connection type.

1. Create a new Project for your data and configure the data load. For example, for file-based data, use the File Data tab, or for relational data, use the Relational Data tab.

2. In the Project view, display the Mapping or Column Mapping dialog:
   - If you’re loading file-based data, click File Targets, then click Column Mapping.
   - If you’re loading from a relational data source or a pluggable data source, then click Mapping.

3. Select a column to edit.

4. Click the Target Expression to display the Expression dialog,

5. Click Default, and select %%SURROGATE_KEY.

You can now use this key to improve the performance of your reporting queries when there is more than one natural key column, or when the natural key is a variable character (’varchar’) column.

Transforming Your Data Using Joins

With Data Sync, you can use joins to transform and cleanse your relational data.

For example, you can de-normalize data, resolve foreign keys based on natural keys, or perform a calculation based on values in a different table.
1. Create a new Project for your data and configure the data load. For example, for file-based data, use the File Data tab, or for relational data, use the Relational Data tab.

2. In the Project view, display the Mapping or Column Mapping dialog:
   • If you’re loading file-based data, click File Targets, then click Column Mapping.
   • If you’re loading from a relational data source or a pluggable data source, then click Mapping.

3. If required, add additional columns to the target table.

4. Click Joins.
   Use the Joins dialog to create and manage joins for the current project.

1. On the Joins dialog, click New, and define the following:
   • Use the Name field to specify a short user-friendly name to identify the join in Data Sync.
   • Use the Table Names field to specify the names of the tables to join, separated by commas.
     If the tables being looked–up are populated by the same job, Data Sync populates the lookup tables before running this data flow.
   • Use the Join field to specify a SQL command that creates the join.
     You can join more than one table in a join statement (in the ANSI SQL style). You can also define aliases for the tables that you’re joining. When defining an alias, make sure that the expression for the columns is specified as alias.columnName. The base table is a runtime stage table, therefore you must prepend the table name with %%. For example, if we are loading ORDER table with a join to PRODUCT table, the join condition might be:

   INNER JOIN PRODUCT ON %%ORDER.PRODUCT_ID = PRODUCT.PRODUCT_ID
   Or:
   LEFT OUTER JOIN PRODUCT ON %%ORDER.PRODUCT_ID = PRODUCT.PRODUCT_ID

2. If the join returns more than one possible match, then use an aggregate function while referring to a column from this join statement.
   If the join does result in multiple matches, check the “Yields Multiple Matches” property.

1. Add columns to the target table (click Target Tables/DataSets, then Table Columns) with the appropriate data type.

2. On the Column Mappings dialog or Mapping dialog, click Unmapped Columns.
3. On the Choose Columns dialog, move the new columns to the Selected Columns list and click OK.

4. For the new column, click Target Expression, choose the lookup being used, and in the expression specify a valid expression referring to any column from this join or any of the base columns.

If the lookup yields multiple matches, use a valid aggregate function such as MIN, MAX, COUNT, AVG etc. For example MIN(PRODUCT.PRODUCT_NAME).

Tracking Information About Your Data

Use Data Sync to store information about your source data.

For example, you can record the date on which data was loaded.

Note:

You can use Data Sync to transform relational data if you're loading data into either Database As A Service or an on-premises database that is configured using the 'Oracle (Thin)' connection type.

1. Create a new Project for your data and configure the data load.

For example, for file-based data, use the File Data tab, or for relational data, use the Relational Data tab.

2. In the Project view, display the Mapping or Column Mapping dialog:
   • If you're loading file-based data, click File Targets, then click Column Mapping.
   • If you're loading from a relational data source or a pluggable data source, then click Mapping.

3. Click the Target Expression to display the Expression dialog.

4. Click Default, and select the data you want to track:
   • UPSERT_TIMESTAMP – Track the date and time of the data load.
   • DML_CODE – Track the update type, that is 'I' for insert or 'U' for update.

Manipulating Your Data Before And After Data Loads

Use Data Sync to apply SQL logic before or after each data load.

For example, to improve data load performance, you might create a table index before you start the data load.

Note:

You can use Data Sync to transform relational data if you're loading data into either Database As A Service or an on-premises database that is configured using the 'Oracle (Thin)' connection type.
1. In the Project view, select your data loading project.

2. Click **Pre/Post SQL Processing**, then click **New**.

3. On the Edit tab, specify the following details.

<table>
<thead>
<tr>
<th>Field or element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Specify a short name to identify the SQL processing operations in the Data Sync client tool and in log files.</td>
</tr>
<tr>
<td>SQL(s)/Stored Procedure(s)</td>
<td>Display the SQL(s)/Stored Procedure(s) dialog where you define your SQL statements and functions:</td>
</tr>
<tr>
<td>Pre/Post</td>
<td>Choose <strong>Pre</strong> to execute the SQL code before each data load. Choose <strong>Post</strong> to execute the SQL code after each data load.</td>
</tr>
<tr>
<td>Inactive</td>
<td>Activate or inactivate the process.</td>
</tr>
</tbody>
</table>

4. (Optional) Click SQL Source Tables, and specify source tables to identify the tables that SQL statements read from.

   This helps to optimize the overall execution time. If you don’t specify source tables, Data Sync defers the step until all extraction tasks are complete in the current project.

5. (Optional) Click SQL Target Tables, and specify target tables to edit tables before SQL execution.

   In addition, this tab enables you to specify when to truncate a table (similar to providing a load strategy).
This part explains how to model data.

Chapters:

• Understanding Data Modeling
• Starting to Build Your Data Model
• Defining Hierarchies and Levels for Drilling and Aggregation
• Securing Your Data Model
Understanding Data Modeling

You build a model of your business data to enable analysts to structure queries in the same intuitive fashion as they ask business questions.

Topics:
- About Modeling Data
- Planning a Data Model

About Modeling Data

A data model is a design that presents business data for analysis in a manner that reflects the structure of the business. Data models enable analysts to structure queries in the same intuitive fashion as they ask business questions. Well-designed models are simple and mask the complexity of the underlying data structure.

Using Data Modeler you can model data from various source types, such as star and snowflake, in various ways that make sense to business users. You must have the BI Data Model Author role to use Data Modeler.

Although not all source objects have star relationships, Data Modeler presents data as a simple star structure in the data model. In other words, the data model represents measurable facts that are viewed in terms of various dimensional attributes.

When building a data model, you perform the following tasks:
- Connect to the database containing your business data.
- Add source tables or views to the model and classify them as either a fact table or a dimension table.
- Define joins between fact and dimension tables.
- Ensure that every dimension table maps to at least one fact table, and that every fact table maps to at least one dimension table.
- Specify aggregation rules for different fact columns, create derived measures based on expressions, create dimension hierarchies to support drilling, and create level-based measures.
- Publish your data model to permanently save the changes and make the data available for use in analyses.

Source data from files or relational sources can be uploaded to tables in connected databases. See Connecting to Data in a Database.

After publishing your data model, you can start visualizing your data from your enterprise reporting Home page. Your data model displays as a subject area that you can use in visualizations, reports and dashboards. The name of the subject area matches the name of your data model.

Note that when you model source objects with multiple star relationships, they’re all part of the same data model and are included in the same subject area.
Planning a Data Model

Before you start modeling your data, take some time to think about your business requirements and to understand data modeling concepts.

Topics:

• Understanding Data Model Requirements
• Components of Data Models
• About Modeling Source Objects with Star Relationships
• About Modeling Source Objects with Snowflake Relationships
• About Modeling Denormalized Sources
• About Modeling Normalized Sources

Understanding Data Model Requirements

Before you can begin to model data, you must first understand your data model requirements:

• What kinds of business questions are you trying to answer?
• What are the measures required to understand business performance?
• What are all the dimensions under which the business operates? Or, in other words, what are the dimensions used to break down the measurements and provide headers for the reports?
• Are there hierarchical elements in each dimension, and what types of relationships define each hierarchy?

After you have answered these questions, you can identify and define the elements of your business model.

Components of Data Models

Fact tables, dimension tables, joins, and hierarchies are key components you will come across when building your data model.

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fact Tables</td>
<td>Fact tables contain measures (columns) that have aggregations built into their definitions. Measures aggregated from facts must be defined in a fact table. Measures are typically calculated data such as dollar value or quantity sold, and they can be specified in terms of hierarchies. For example, you might want to determine the sum of dollars for a given product in a given market over a given time period. Each measure has its own aggregation rule such as SUM, AVG, MIN, or MAX. A business might want to compare values of a measure and need a calculation to express the comparison.</td>
</tr>
</tbody>
</table>
Component | Description
---|---
Dimension Tables | A business uses facts to measure performance by well-established dimensions, for example, by time, product, and market. Every dimension has a set of descriptive attributes. Dimension tables contain attributes that describe business entities (like Customer Name, Region, Address, or Country).

Dimension table attributes provide context to numeric data, such as being able to categorize Service Requests. Attributes stored in this dimension might include Service Request Owner, Area, Account, or Priority.

Dimension tables in the data model are conformed. In other words, even if there are three different source instances of a particular Customer table, the data model only has one table. To achieve this, all three source instances of Customer are combined into one using database views.

Joins | Joins indicate relationships between fact tables and dimension tables in the data model. When you create joins, you specify the fact table, dimension table, fact column, and dimension column you want to join. Joins allow queries to return rows where there is at least one match in both tables.

Tip: Analysts can use the option **Include Null Values** when building reports to return rows from one table where there no matching rows in other table.

See Suppressing Null Values in Views in *Using Oracle Business Intelligence Cloud Service*.

Hierarchies | Hierarchies are sets of top-down relationships between dimension table attributes.

In hierarchies, levels roll up from lower levels to higher levels. For example, months can roll up into a year. These rollups occur over the hierarchy elements and span natural business relationships.

About Modeling Source Objects with Star Relationships

Star sources consist of one or more fact tables that reference any number of dimension tables. Because Data Modeler presents data in a star structure, working with star sources is the simplest modeling scenario. In star sources, dimensions are normalized with each dimension represented by a single table.

For example, assume that you have separate sources for Revenue Measures, Products, Customers, and Orders. In this scenario, you load data from each source to separate database tables. Then, you use Data Modeler to create a fact table (Revenue Measures) and dimension tables (Products, Customers, and Orders). Finally, you create joins between the dimension tables and the fact table.

When you create your fact and dimension tables, you can drag and drop the source objects into the data model, or you can use menu options to create the fact and dimension tables individually.

See Roadmap for Modeling Data for a full list of data modeling tasks.
About Modeling Source Objects with Snowflake Relationships

Snowflake sources are similar to star sources. In a snowflake structure, however, dimensions are normalized into multiple related tables rather than in single dimension tables.

For example, assume that you have separate sources for Revenue Measures, Products, Customers, and Orders. In addition, you have separate sources for Brands (joined to Products) and Customer Group (joined to Customers). The Brands and Customer Group tables are considered to be “snowflaked” off the core dimension tables Customers and Products.

In this scenario, you load data from each source to separate database tables. Next, you create database views that combine the multiple dimension tables into a single table. In this example, you create one view that combines Products and Brand, and another view that combines Customer and Customer Group.

Then, you use Data Modeler to create a fact table (Revenue Measures) and dimension tables (Products + Brand view, Customers + Customer Group view, and Orders). Finally, you create joins between the dimension tables and the fact table.

See Roadmap for Modeling Data for a full list of data modeling tasks.

About Modeling Denormalized Sources

Denormalized sources combine facts and dimensions as columns in one table (or flat file). With a denormalized flat source, one data file is loaded into one table. The data file consists of dimension attributes and measure columns.

In some cases, the data model might consist of a hybrid model that involves a combination of star, snowflake, and denormalized sources. For example, a denormalized source might include information about revenue measures, products, customers, and orders - but all in a single file rather than in separate source files.

In this scenario, you first load the denormalized file as a single database table. Then, you use the Add to Model wizard to partition columns into multiple fact and dimension tables. In this example, you drag and drop revenue measure columns to create a fact table, then drag and drop columns for products, customers, and orders to create three separate dimension tables. Finally, you create joins between the dimension tables and the fact table.

See Roadmap for Modeling Data for a full list of data modeling tasks.

About Modeling Normalized Sources

Normalized or transactional sources distribute data into multiple tables to minimize data storage redundancy and optimize data updates. In a normalized source, you have multiple data files that correspond to each of the transactional tables. Data from Oracle Cloud applications is likely partitioned into a normalized source.

Similar to snowflake sources, modeling normalized sources involves creating database views to combine columns from multiple source tables into individual fact and dimension tables. Some normalized sources are very complex, requiring a number of database views to organize the data into a star-type model.
For example, assume that you have source files for Products, Customers, Orders, and Order Items. Orders and Order Items both contain facts.

In this scenario, you first load the files as separate database tables. Next, you create a database view that combines the multiple fact columns into a single table. In this example, you create a view that combines columns from Orders and Order Items.

Then, you use Data Modeler to create a fact table (Orders + Order Items view) and dimension tables (Products and Customers). Finally, you create joins between the dimension tables and the fact table.

See Roadmap for Modeling Data for a full list of data modeling tasks.
Starting to Build Your Data Model

This section provides information about first steps for building a data model, such as adding dimension tables, fact tables, and joins.

Topics:

- Typical Workflow for Modeling Data
- Using Data Modeler
- Reviewing Source Tables and Data
- Adding Your Own Source Views
- Adding Fact Tables and Dimension Tables to the Data Model
- Joining Fact and Dimension Tables
- Creating a Time Dimension
- Using Columns in the Data Model
- Copying Model Objects

Typical Workflow for Modeling Data

Here are the common tasks for modeling data.

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>More Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read about Data Modeler</td>
<td>Get familiar with Data Modeler, including how to refresh your data, publish changes, and find the Action menus.</td>
<td>Using Data Modeler</td>
</tr>
<tr>
<td>Create a new model</td>
<td>Start a new model and connect it to your data source.</td>
<td>Creating a Data Model</td>
</tr>
<tr>
<td>Browse source objects</td>
<td>Review source tables to determine how to structure your data model.</td>
<td>Reviewing Source Tables and Data</td>
</tr>
<tr>
<td>Create new views in the database if needed</td>
<td>Create views for role-playing dimensions, or create views to combine multiple tables into a single view, as in snowflake or normalized sources.</td>
<td>Adding Your Own Source Views</td>
</tr>
<tr>
<td>Add fact tables and dimension tables</td>
<td>Create fact tables and dimension tables from source objects.</td>
<td>Adding Fact Tables and Dimension Tables to the Data Model</td>
</tr>
<tr>
<td>Join fact and dimension tables</td>
<td>Create joins between fact and dimension tables.</td>
<td>Joining Fact and Dimension Tables</td>
</tr>
</tbody>
</table>
### Task

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>More Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add a time dimension</td>
<td>Create a time dimension table and database source table with time data.</td>
<td>Creating a Time Dimension</td>
</tr>
<tr>
<td>Add aggregated and calculated measures</td>
<td>Specify aggregation for columns and create calculated measures using expressions.</td>
<td>Specifying Aggregation for Measures and Creating Calculated Measures</td>
</tr>
<tr>
<td>Add derived attributes</td>
<td>Specify custom attributes for dimension tables using expressions.</td>
<td>Creating Derived Attributes</td>
</tr>
<tr>
<td>Create hierarchies and levels</td>
<td>Define hierarchies and levels based on relationships between groups of attribute columns.</td>
<td>Editing Hierarchies and Levels</td>
</tr>
<tr>
<td>Create variables</td>
<td>Optionally, create variables that dynamically calculate and store values for use in column expressions and data filters</td>
<td>Defining Variables</td>
</tr>
<tr>
<td>Set up object permissions</td>
<td>Control who can access fact tables, dimension tables, and columns.</td>
<td>Securing Access to Objects in the Model</td>
</tr>
<tr>
<td>Set up data security filters</td>
<td>Define row-level data security filters for fact tables, dimension tables, and columns.</td>
<td>Securing Access to Data</td>
</tr>
</tbody>
</table>

## Using Data Modeler

Data Modeler enables you to model the data that is needed to produce reports.

**Topics:**

- Opening Data Modeler
- Creating a Data Model
- Using the Left Pane in Data Modeler
- Using the Right Pane in Data Modeler
- Using Action Menus
- Locking a Data Model
- Validating a Data Model
- Refreshing and Synchronizing Source Objects and Data Model Objects
- Publishing Changes to a Data Model
- Clearing Cached Data
- Renaming a Data Model
- Connecting a Model to a Different Database
- Exporting a Data Model
- Importing a Data Model
- Deleting a Data Model
Opening Data Modeler

You need the BIDataModelAuthor role to use Data Modeler. Ask your administrator to give you access if you don’t see this option.

1. Sign in to Oracle BI Cloud Service.
2. Click **Data Sources** on the Home page.

3. Click **Manage Models** in the Create section.
4. Click the name of a model to open it in Data Modeler.

5. To start a new model, click **Create model**.

Creating a Data Model

Create a new data model from scratch in Data Modeler.

1. Open Data Modeler.
2. Click **Create model**.
3. Enter a name and description for your data model.
   The subject area associated with this model gets the same name.
4. Connect the model to a **Database**.
   If the database you want isn't listed, ask your administrator to set up the connection for you. See **Managing Database Connections**.

Using the Left Pane in Data Modeler

Various data modeling menus are available from the left pane in Data Modeler.

- **Database** — Lists source objects such database tables and views
- **Data Model** — Lists data model objects such as fact tables, dimension tables, hierarchies, fact columns, and dimension columns
- **Variables** — Lists variables for use in data security filters and in column expressions
• **Roles** — Lists roles that you can use when defining object permissions and data security filters

Filter a list to find exactly what you want.

1. In Data Modeler, in the left pane, open the **Database, Data Model, Variables, or Roles** menu.
2. Click the **Filter** icon to the right of the selected menu.
3. In the Filter area, enter a string value for filtering the display.
4. Delete the text or click the **Filter** icon again to remove the filter.
Using the Right Pane in Data Modeler

The right pane in Data Modeler is a contextual pane that changes depending on what task you’re performing. After you have started modeling data, the default or home view shows the fact tables, dimension tables, and joins that you’ve defined so far.

- In the fact tables and dimension tables area you can see the number of joins for each fact and dimension table, as well as the number of measures in each fact table.

- Joins are listed below the fact and dimension tables. Click the up or down arrow in each column header to sort.

- When you click an object to open its editor, the editor appears in the right pane. For example, clicking a dimension table name from the Data Model menu in the left pane opens the dimension table editor in the right pane.
• Open the Permission tab to control who has access to the model and who is allowed to build reports from its associated subject area.

• Open the Properties tab to rename the model or connect the model to a different database.

Using Action Menus

Data Modeler provides action menus for most objects. Action menus are represented by a gear icon (⚙️).

Action menus contain actions that are relevant for a particular object or context, and are visible when the object is selected. For example, select a source object in the Database menu in the left pane to see its action menu.

A global Model Actions menu is also provided in the upper right corner. You use the global Model Actions menu for tasks that apply to the entire data model, such as clearing, closing, refreshing, or unlocking the model.

Deleting Individual Data Model Objects in Data Modeler

You can use action menus to delete data model objects. Note the following about deleting objects:

• You must lock the model to delete an object.

• You can delete source views but you can't delete source tables. Use SQL Workshop to drop tables in the source database.

• You can't delete model objects that other objects depend on. For example, you can't delete a dimension table that is joined to another table until the join is removed. Similarly, you can't delete a column that's used in an expression, or a source view that's being used in another view.

• Objects aren't truly deleted until changes are published, with the exception of source views. Source views are deleted when you complete the action.

Locking a Data Model

You must always lock the data model before making any changes. Click Lock to Edit to lock the data model.
Always publish changes you want to keep before leaving Data Modeler for an extended length of time. When your HTTP browser session times out (after 20 minutes of inactivity), the lock is released and any unpublished changes are discarded.

Similarly, closing a browser ends the HTTP session and discards any unpublished changes. However, closing the browser does not release the lock. In this case, you can start a new session in a new browser and sign in with the same user name. When you attempt to lock the model in the new session, Data Modeler asks whether you want to reacquire the lock.

**Changing Database Views**

You must also lock the model if you want to change database views from Data Modeler. Changes you make to database views are immediately saved to the database. This is different to data model changes which are only saved when you publish them.

Locking the model prevents other users from changing database views using Data Modeler. The lock does not stop someone from modifying database objects using other tools, such as APEX and SQL Developer.

**Overriding Locks**

If you have administrative privileges, you can override locks set by other users. To do this, select **Override Lock** from the global Model Actions menu in the upper right corner. Overriding a lock discards changes made by other users in their browser sessions. You must have the BIServiceAdministrator role to override a lock.

**Validating a Data Model**

You can use the global **Validate** checkmark icon in the upper-left corner to check whether a data model is valid.

The data model is also validated automatically when you publish changes. Validation errors are shown at the bottom of the right pane.

Use the **Message Actions** menu to customize the types of messages displayed (Errors, Warnings, and Information).

Some tasks are validated when they’re performed. For example, you can’t save a source view unless its SQL query is valid. Expressions for calculated measures and derived columns must be valid before they can be saved. Validation messages that are displayed as you’re performing tasks provide more information about any validation errors.
Refreshing and Synchronizing Source Objects and Data Model Objects

Data Modeler provides three ways to refresh data to ensure you’re looking at the most up-to-date information. You can refresh source objects, refresh the data model, or synchronize the data model with source object definitions in the database.

**Refreshing Source Objects**

You can refresh the Database pane to ensure that the source objects list reflects the latest objects in the database. For example, you can refresh the source objects list to include any new database tables that were added. The source objects list is not refreshed automatically after new objects are loaded into the database.

To refresh source objects, select **Refresh** from the **Database Actions** menu in the left pane.

![Refreshing Source Objects](image)

**Refreshing the Data Model**

In some cases, other Data Modeler users might have locked the model and made changes. You can refresh the data model to ensure that Data Modeler is displaying the latest version of the model.

To refresh the data model, select **Refresh** from the **Data Model Actions** menu in the left pane.

Alternatively, select **Refresh Model** from the **Model Actions** gear menu next to the **Lock to Edit** button.

**Synchronizing with the Database**

You can synchronize the data model with source objects in the database. Synchronization identifies objects in the model that have been deleted in the database, as well as tables and columns that are new. It also identifies other discrepancies like column data type mismatches.

To synchronize all model objects and source objects with the database, select **Synchronize with Database** from the global **Model Actions** menu in the upper right corner.

To synchronize individual fact tables or dimension tables, select **Synchronize with Database** from the **Actions** menu for the given fact table or dimension table in the Data Model objects list in the left pane. Then, click **OK**.

You must lock the data model to synchronize with the database.
Synchronization discrepancies are displayed in a message box at the bottom of the right pane. Use the **Message Actions** menu to customize the types of messages displayed (Errors, Warnings, and Information), select or deselect all messages, and perform sync-up actions on selected messages. For example, you can select all data type mismatch warnings and then select **Sync-up selected** from the **Actions** menu to make the relevant synchronization changes.

**Publishing Changes to Your Data Model**

As you update a data model, you make changes that you can save or discard. You publish a model to save the changes permanently and make the data available for use in reports. The published data model displays as a subject area.

**Tip:**

Although changes to the data model are saved as you work, they are saved in the browser session only. The changes aren’t truly saved until you publish the model.

When you publish a data model, it is validated automatically. Any validation errors appear in the bottom of the right pane. If you see validation errors, fix them and then try to publish the data model again.

After making changes to your data model, you can perform these actions using the menus in the upper-right corner:
• **Publish and Unlock** — Verifies that the model is valid, saves the changes, and publishes the model for use with reports. The model is unlocked for other users.

• **Publish and Keep Lock** — Verifies that the model is valid, saves the changes, and publishes the model for use with reports. The lock is retained for further edits.

• **Unlock** — Removes the lock on the model so that other users can update it. Your unpublished changes to the model are discarded.

• **Revert** — Returns the model to its previously published state. Your unpublished changes to the model are discarded, but the model remains locked.

• **Clear**—Permanently deletes all objects in the model and removes them from any reports that are based on the model’s subject area.

You can also click **Undo** and **Redo** in the upper right corner to revert or reapply individual changes.

**Tip:**

You don’t need to publish the model to save database changes. Changes made to database views and other source database objects are saved to the database when you complete the action, not to the data model. For database changes, **Undo** and **Redo** aren’t available.

After publishing your model it takes up to two minutes for changes to the data model to reflect in reports and dashboards. To see changes immediately, open the report, click **Refresh**, and then **Reload Server Metadata**.

Oracle BI Cloud Service takes a snapshot when you or someone else publishes changes to the data model. If you’re having some problems with the latest data model, you can ask your administrator to restore an earlier version. See **Restoring from a Snapshot**.

**Clearing Cached Data**

Oracle BI Cloud Service caches data to maximize performance. This means data updates may not immediately reflect in reports and Data Modeler.

After loading new data in your tables, you might want to clear the cache to see the very latest data.

• To see new data in Data Modeler, select the **Refresh Model** menu.

• To see new data in reports, manually clear the cache from the Data Model menu in the left pane
  
  – To clear cached data for a particular fact or dimension table, right-click the table and select **Clear Cached Data**.

  – To clear all cached data, click **Data Model Actions**, then select **Clear All Cached Data** to remove all data from the cache.

You can also select **Clear All Cached Data** from the global **Model Actions** menu in the upper-right corner.
Renaming a Data Model

To rename a data model, lock it, select the Properties tab, and change the name. This action also renames the corresponding subject area for reports.

Connecting a Model to a Different Database

When you start a new data model you’re asked to select the database where your data is stored. All the tables and views in this database display in Data Modeler so you can add them to your model. Sometimes, data is moved or the source database changes. If this happens, change your model’s database connection.

Note:

If you change the database, reports based on the model's subject area won't work unless all the required source objects are available in the new database.

1. In Data Modeler, lock your model for editing.
2. Click the Properties tab.
3. Select the Database.

If the database you want isn't listed, ask your administrator to set up the connection for you. See Managing Database Connections.
4. Synchronize your data model with the new database. Select Synchronize with Database from the Model Actions menu.

See also, Refreshing and Synchronizing Source Objects and Data Model Objects.

Exporting a Data Model

Individual data models can be exported to a JSON file and the information imported on another service. If you want to make minor changes to the model, you can edit the JSON before importing it. For example, you might want to change the name of the model (modelDisplayName) or the database connection (connectionName).

1. Open Data Modeler.
2. In the Models page, click the Model Actions icon for the model you want to export, and select Export.
3. Save the JSON file. The default name is model.json.

Importing a Data Model

Individual data models can be exported to a JSON file and the information imported on another service. If you want to make minor changes to the model, you can edit the JSON before importing it. For example, you might want to change the name of the model (modelDisplayName) or the database connection (connectionName).

For any data model to work properly it must have access to the associated database tables. Before importing the data model, check whether Data Modeler can connect to the required database. If not, ask your administrator to set up the connection. See Connecting to Data in an Oracle Cloud Database.

1. Open Data Modeler.
2. Click Import Model.
3. Browse to the JSON file that contains the data model you want to import.
4. Click OK.
5. Optional: Select a database connection for the model.
You’re asked to select a database connection if Data Modeler doesn’t recognize the connection name in the JSON file. If the connection you want isn’t listed, ask your administrator to set up the connection and try again.

6. Optional: Choose whether to replace a data model with the same name. Click Yes to overwrite the model or No to cancel.

This happens when the model named in the JSON file clashes with another model in Data Modeler. If you don’t want to replace the existing model, change the modelDisplayName attribute in the JSON file and try again.

Deleting a Data Model

You can delete all objects from your data model if you want to clear your model and start over. Or you can delete an entire model along with its subject area.

• Clearing model content—Lock the model and select Clear Model from the global Model Actions menu in the upper right corner.
  
  This permanently removes all the objects in the data model and also removes them from any reports that are based on the model’s subject area.

• Deleting a model—Click Data Modeler, click the Model Actions menu for the model you don’t want anymore, and select Delete.
  
  This permanently removes the data model and its subject area.

Before clearing or deleting a model, we recommend that you or your administrator take a snapshot of the model as a backup. For instructions, see Taking Snapshots and Restoring.

Reviewing Source Tables and Data

This topic describes how you can learn more about the source database objects that are available for your data model.

Topics:

• Viewing Source Objects
• Previewing Data in Source Objects
Viewing Source Objects

You can see a list of source tables and views in the Database menu in the left pane. Click a table or view to see its properties.

The Overview tab for source tables and views shows column information, like column name, data type, whether it's unique, and whether it accepts null values. See Previewing Data in Source Objects.

Previewing Data in Source Objects

You can preview the first 25 rows of data in your database tables and views. By reviewing the initial rows, you can get ideas for modeling the database tables and views as either dimension tables or fact tables.

1. Open Data Modeler.
2. From the Database menu in the left pane, click a database table or view to open it.
3. Click the Data tab.
4. Review the first 25 rows of data for the table or view. You can resize the columns in the display table if needed.
5. Click Get Row Count to retrieve a complete row count for the table or view. This take might take some time to complete if the table is large.
6. Click Done.

Creating Source Views

Create source views as a base for model objects when you think you might want to perform subsequent changes.

Topics:
- About Source Views
- Defining Source Views
- Defining Filters for Source Views

About Source Views

Source views are saved queries of data in the database. You can think of a source view as a "virtual table."

You create source views when using a single table as a source for more than one dimension table. For example, you can create source views that use the Employee source table as a source for the Employee and Manager dimension tables.

You also create source views when creating a dimension table that is based on multiple source tables, as in a snowflake source. For example, you can create a source view that combines columns from the Customer and Customer Group source tables to create a single Customers dimension table.

You can also perform pre-aggregation calculations in a source view. For example, to create an Average Revenue column that is calculated pre-aggregation, you can include the calculation in the SQL query for the view:

```
SELECT
    "BICS_REVENUE_FT1"."UNITS",
    "BICS_REVENUE_FT1"."ORDER_KEY",
    "BICS_REVENUE_FT1"."REVENUE",
    "BICS_REVENUE_FT1"."PROD_KEY",
    "BICS_REVENUE_FT1"."REVENUE"/"BICS_REVENUE_FT1"."UNITS" AS AVERAGE_REVENUE
FROM
    "BICS_REVENUE_FT1"
```

In general, create source views as a base for model objects when you think you might want to perform subsequent changes. Creating a data model based on source views provides greater flexibility than using source tables directly. For example, using source views makes it much easier to extend model objects, create filters, and add pre-aggregation calculations.

Adding Your Own Source Views

You can add views to the source database from Data Modeler. For example, you can create a source view that combines the Brands and Products source tables to create a single source for your dimension table.

Create source views as a base for model objects when you think you might want to perform subsequent changes. You can create a view from scratch and add any
column you want from other tables and views in the database. Alternatively, you can create a view by copying an existing source table or another source view.

1. In Data Modeler, lock the model for editing.

2. From the Database menu in the left pane, click **Actions**, then click **Create View**.

Initially the view is empty. You can add any column you want from other tables and views in the database.

**Tip:**

To create a view from an existing source table or source view, navigate to the database object you want to copy, click **Actions**, and then click **Duplicate**. See also Copying Model Objects.

3. In the View editor, specify a name and description for the view. Optionally deselect **Remove duplicate rows** if you want to include duplicate rows in the view.

4. Add columns to the database view by dragging and dropping tables or views from the Database menu into the Columns area of the View editor.

Alternatively, click **Add Columns**, select a source database table or view, select columns, and then click **Add**.
5. Define aliases for columns if needed. You can also optionally move rows up or down using the Action menu for a specific row.

6. From the Joins tab, you can define joins for the view. Click Create Join, then specify the left side table, right side table, columns, and the join type. You must include more than one source table in your view to create joins.

7. From the Filters tab, you can define filters for the view. See Defining Filters for Source Views.

8. From the SQL Query tab, review the code for the SQL query for the source view.

You can edit the SQL code for the query here, but do so only if you’re familiar with SQL code. Entering invalid SQL code can produce unexpected results.

If you do edit the SQL query directly, simple updates are reflected back in the Overview, Join, and Filters tabs and you can use these tabs to further edit the view later. For example, you can include:

- Simple SELECT clause with aliases and DISTINCT keyword
- FROM clause with joins
- WHERE clause with filter conditions which combined with AND keyword

If you use the SQL Query tab to make more advanced code changes you cannot use the Overview, Joins or Filters tabs to further edit the view. For example, if you include:

- SQL aggregation functions, GROUP BY clause, HAVING clause
- ORDER BY clause
- OR keyword in WHERE clause

9. Optionally, click the Data tab to preview the first 25 rows of data. You can also get a complete row count. It is best to view data only after defining joins between all tables for better performance.

10. Click Save and Close.

Defining Filters for Source Views

A filter specifies criteria that are applied to columns to limit the results that are returned. In other words, a filter is the WHERE clause for the view statement. For example, you can define a filter where Customer Country is equal to USA.
1. Create a view. See Adding Your Own Source Views.
2. Click the Filters tab.
3. Click Create Filter.
4. In the WHERE row, first select the column for the filter. Next, select the condition, such as "is not equal to" or "is greater than". Finally, specify the value for the filter. You can specify a variable if needed.

5. Optionally, click Create Filter again to add an "and" row to the filter. Specify the column, condition, and value. Repeat as needed.
6. To remove a row, click Actions, then select Delete.

7. Click Save.

Adding Fact Tables and Dimension Tables to a Data Model

Use fact tables and dimension tables to represent aspects of your business that you want to understand better.

Topics:
- About Fact Tables and Dimension Tables
- Creating Fact and Dimension Tables from a Single Table or View
- Creating Fact Tables Individually
- Creating Dimension Tables Individually
- Editing Fact Tables and Dimension Tables
- Adding More Columns to Fact and Dimension Tables

About Fact Tables and Dimension Tables

Fact tables and dimension tables hold the columns that store the data for the model:
- Fact tables contain measures, which are columns that have aggregations built into their definitions. For example, Revenue and Units are measure columns.
- Dimension tables contain attributes that describe business entities. For example, Customer Name, Region, and Address are attribute columns.
Fact tables and dimension tables represent the aspects of your business that you want to understand better. See Components of Data Models.

Before you begin modeling fact tables and dimension tables, make sure that the data that you need to model is available in the source tables list. Also ensure that you have created any source views upon which to base model objects.

If you think the list of source objects in the database has changed since you opened Data Modeler, then you can click Refresh from the Database Actions menu. If the data that you need has not yet been loaded into the database, then you can load it.

Creating Fact and Dimension Tables from a Single Table or View

Some source tables contain both facts and dimensions. For these source tables, Data Modeler provides a wizard to help you partition the fact and dimension columns into fact tables and dimension tables.

Video

For example, you might have a source that contains both product and customer attributes, as well as revenue measures. Use the wizard to create the corresponding fact and dimension tables.

1. In Data Modeler, lock the model for editing.
2. In the Database menu in the left pane, right-click the source table that contains the fact and dimensional data that you want to model, select Add to Model, and then select Add as Fact and Dimension Tables.
3. To let Data Modeler suggest some fact tables, dimension tables, and joins for the source table, select Let Data Modeler Recommend and click OK. You can review suggestions in Step 4.

If you'd rather choose fact and dimension tables yourself from scratch:

a. Deselect Let Data Modeler Recommend and click OK.
b. Drag measures from the source table onto the fact table.
Tip:
You can also click the **Plus** icon in the column header area to select a column to include in the fact table.

c. Enter a name for the fact table, such as Costs or Measures.
d. Add a dimension table for each group of related attributes, and enter a meaningful name, such as Products. Drag and drop related columns from the source table to the appropriate dimension table.

e. To add more dimension tables, click **Add** and repeat the previous step.
f. To delete a dimension table, click **X** next to the table name.
g. Specify the join columns for each of the dimension tables. Select the box beside the appropriate columns to mark those columns as join columns.

If the join column you select is missing from the fact table, a corresponding column gets added automatically to the fact table.

4. Review fact tables, dimension tables, and join columns. For example:
   - Rename fact and dimension tables.
   - Add or remove columns.
   - Add, delete, or merge dimension tables.
   - Move columns from one dimension table to another.

5. Click **Next**.
6. Review the objects that will be created.

7. Click **Create**.

8. Click **Done**.

New fact tables, dimension tables, and joins display in Data Modeler. New views display in the Database pane.

Creating Fact Tables Individually

You can add individual source tables containing fact data to your data model.

If you have distinct source tables with fact data, such as in a star source, then you can add them to your data model individually. For example, if you have a source table that contains only revenue measures, then you can use this method to create the corresponding fact table.

Alternatively, you might have sources with fact information spread across multiple tables, such as normalized transactional sources. In this case, create source views first to combine tables in a way that resembles a star model. For information about creating views, see Defining Source Views. For information about modeling different source types, see Planning a Data Model.

Tip:

Create source views as a base for model objects when you think you might want to perform subsequent changes like extending model objects, creating filters, and adding pre-aggregation calculations. Creating a fact table based on source views provides greater flexibility than using source tables directly.

When you use this method to create individual fact tables, all columns in the source table or view are assigned to a single fact table and if the source has relationships with other tables or views, we'll offer to add them to your model.

After locking the model, perform one of the following actions to create fact tables individually:

- Drag the source table or view from the Database menu in the left pane to the Fact Tables area of the Data Model.
• From the Database menu in the left pane, right-click the table or view, then click **Add to Model**, then **Add as Fact Table**.

• From the Database menu in the left pane, click **Table Actions** or **View Actions**, click **Add to Model**, then **Add as Fact Table**.

• From the Database Table or View editor for a particular source table or view, click **Add to Model**, then **Add as Fact Table**.

• In the right pane, click **Add** in the Fact Tables area of the Data Model. Then, select one or more source tables and views from the Database Objects list and click **OK**.

• To copy an existing fact table, click **Fact Table Actions** for the fact table you want to copy, and then click **Duplicate**. See also **Copying Model Objects**.

After adding the source table or view to the model, you can edit the fact table. See **Editing Fact Tables and Dimension Tables**.

### Creating Dimension Tables Individually

You can add individual source tables containing dimension data to your data model.

If you have distinct dimensional source tables, such as in a star source, then you can add them to your data model individually. For example, if you have a source table that contains only customer attributes, then you can use this method to create the corresponding dimension table.

Alternatively, for snowflake or normalized (transactional) sources, create source views to combine source objects in a way that resembles a star model. For information about creating views, see **Defining Source Views**. For information about modeling different source types, see **Planning a Data Model**.

**Tip:**

Create source views as a base for model objects when you think you might want to perform subsequent changes like extending model objects, creating filters, and adding pre-aggregation calculations. Creating a dimension table based on source views provides greater flexibility than using source tables directly.

When you use this method to create individual dimension tables, all columns in the source table or view are assigned to a single dimension table and if the source has relationships with other tables or views, we’ll offer to add them to your model.

After locking the model, perform one of the following actions to create dimension tables individually:

• Drag the table or view from the Database menu in the left pane to the Dimension Tables area of the Data Model.

• From the Database menu in the left pane, right-click the table or view, click **Add to Model**, and then select **Add as Dimension Table**.

• From the Database menu in the left pane, click **Table Actions** or **View Actions** for a table or view, click **Add to Model**, and then select **Add as Dimension Table**.
• Click **Add** in the Dimension Tables area, and then select **Add Database Tables**. From the Database Objects list, select one or more sources and then click **OK**.

• From the Database Table or View editor for a particular source table or view, click **Add to Model** and then select **Add as Dimension Table**.

• To copy an existing dimension table, click **Dimension Table Actions** for the dimension table you want to copy, and then click **Duplicate**. See also **Copying Model Objects**.

After adding the source table or view to the model, you can edit the dimension table. See [Editing Fact Tables and Dimension Tables](#).

### Editing Fact Tables and Dimension Tables

You can edit properties of fact and dimension tables in your data model and preview the source data.

1. In Data Modeler, lock the model for editing.
2. Click the fact table or dimension table that you want to edit.
3. Change settings on the Overview tab as needed:

<table>
<thead>
<tr>
<th>Field or Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time dimension</td>
<td>For dimension tables only. Specifies that hierarchies for this dimension table support a time dimension.</td>
</tr>
<tr>
<td>Enable skipped levels and Enable unbalanced hierarchies</td>
<td>For dimension tables only. Set properties for hierarchies associated with this dimension table. See <a href="#">Setting Dimension Table Properties for Hierarchies</a>.</td>
</tr>
<tr>
<td>Column list</td>
<td>Click the link for a column to edit that column in the Column editor. Or, right-click the row for the column and click <strong>Edit</strong>. See <a href="#">Editing Columns</a>.</td>
</tr>
<tr>
<td>Field or Element</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>-------------</td>
</tr>
</tbody>
</table>
| Aggregation      | For fact tables only. Click to select a type of aggregation for the column from the list, or select Set Aggregation from the Column Actions menu. Aggregation types include:  
None: Applies no aggregation.  
Sum: Calculates the sum by adding up all values.  
Average: Calculates the mean value.  
Median: Calculates the middle value.  
Count: Calculates the number of rows that aren't null.  
Count Distinct: Calculates the number of rows that aren't null. Each distinct occurrence of a row is counted only once.  
Maximum: Calculates the highest numeric value.  
Minimum: Calculates the lowest numeric value.  
First: Selects the first occurrence of the item.  
Last: Selects the last occurrence of the item.  
Standard Deviation: Calculates the standard deviation to show the level of variation from the average.  
Standard Deviation (all values): Calculates the standard deviation using the formula for population variance and standard deviation. |
| Available        | Click to mark a column as Available or Unavailable to choose whether that column is displayed in analyses that are created. You can also select Mark as Unavailable or Mark as Available from the Column Actions menu. |
| Edit All         | You can click to edit properties for individual columns in the table, or select Edit All to edit all rows at once. |
| Add Column       | Click Add Column to display the Column editor and create a new column. See Editing Columns. |
4. From the Source Data tab, you can preview the first 25 rows of source data for the table. Resize the columns in the display table if needed. Click **Get Row Count** to retrieve a complete row count for the table or view.

5. For dimension tables only: from the Hierarchies tab, edit the hierarchies and levels for the table. See **Editing Hierarchies and Levels**.

6. From the Permissions tab, specify object permissions. See **Securing Access to Objects in the Model**.

7. From the Data Filters tab, you can define data filters that provide row-level filtering for data model objects. See **Defining Data Security Filters**.

8. Click **Done** to return to the data model.

Adding More Columns to Fact and Dimension Tables

There are different ways to add more source columns to fact and dimension tables in your model.

- **Synchronize with the database**
  
  If new columns are added to a source table and you want to include them in fact tables and dimension tables in your model, synchronize the fact or dimension table with the database. Synchronization identifies any new columns and adds them to the fact or dimension table. See **Refreshing and Synchronizing Source Objects and Data Model Objects**.

- **Include columns from another source** (dimension tables only)
  
  Dimension tables can combine columns from multiple sources. See **Adding Columns from Another Source to a Dimension Table**.

Adding Columns from Another Source to a Dimension Table

You can add the columns from another source table or view to an existing dimension table. For example, you may want to include attributes from a Product Category table in your Products dimension table.

1. In Data Modeler, lock the model for editing.
2. Select the dimension table you want to edit so its Overview tab displays.
3. Drag and drop the source table or view that contains the columns you want to add from the Database pane to the dimension table (columns area).
Alternatively, right-click the dimension table you want to edit, click **Add Columns**, and then select the source table or view that contains the columns you want to add.

4. Select appropriate join columns and click **OK**.

![Add Columns to Dimension: Products](image)

View the dimension table to see the additional columns. The Source property shows that the dimension table is based on a new database view. Data Modeler creates a new database view whenever you add columns from another source.

## Joining Tables in a Data Model

A join in the model indicates a relationship between one fact table and one dimension table.

**Video**

**Topics:**
- About Joins
- Joining Fact and Dimension Tables

### About Joins

A join in the model indicates a relationship between one fact table and one dimension table. When you use the Add to Model wizard to model data, the wizard creates joins automatically between a fact table and each of its corresponding dimension tables.

When you model fact and dimension tables individually, joins are automatically created between them if the join references exist in the source tables.

You can also manually create joins in the data model. To do this, you drag and drop a dimension table to a fact table, or click **Create Join** in the Joins area.

When you define a join between a fact table and dimension table, you select a join column from each table. You can create a join on more than one column.
Joining Fact and Dimension Tables

Define joins between fact tables and dimension tables to enable querying of related data. For example, you can define a join between the Profit Metrics fact table and the Products dimension table.

1. In Data Modeler, lock the model for editing.
2. In the Dimensions Tables area, drag and drop a dimension table to the Fact Tables area. Or, in the Joins area, click Create Join.
3. In the Joins area, specify the appropriate Fact Table, Fact Column, Dimension Table, and Dimension Column to use for the join.
   For example, you might specify a billing date column and a calendar date column.
4. Click the checkmark icon to save the changes to the join.
   If you want to remove your changes, then click the X icon. If you start to create a new join and click X, then the new row for the join is removed from the Joins table.

After you create joins, you can see the default hierarchies and levels when you click the Hierarchies tab for the given dimension table.

Creating a Time Dimension

Time series functions provide the ability to compare business performance with previous time periods, enabling you to analyze data that spans multiple time periods. For example, time series functions enable comparisons between current sales and sales a year ago, a month ago, and so on. To use time series functions, the data model must include a time dimension.

Video

When you create a time dimension, the Create Time Dimension wizard creates a table in the database, populates it with time data, creates a corresponding time dimension table in the data model, and creates a time hierarchy.

The Create Time Dimension wizard populates the source table with time data from 01-JAN-1970 to 31-DEC-2020.

1. In Data Modeler, lock the model for editing.
2. In the Dimension Tables area, click Add, then Create Time Dimension.
3. In the Create Time Dimension wizard, specify names for the database table, the dimension table, and the hierarchy.

4. In the Hierarchy Levels, specify which levels to include, such as Year, Quarter, and Month.

5. Click **Next**.

6. On the next page, review the tasks that the wizard will perform to create the time dimension.

7. Click **Create** to enable the wizard to create the dimension.

   The wizard adds a time dimension with data to the database and creates a corresponding dimension in the data model. This action might take up to 30 seconds.

8. Click **Done**.

9. To create joins between columns in the fact table and columns in the Time dimension table, click **Create Join** in the data model.

   The time dimension has two unique columns. The DAY_TS column has the type TIMESTAMP, and the DATE_ID column has the type NUMBER. When you create a join, you specify either the column with the timestamp format or with the numeric format (depending on whether the column in the fact table has a date or number type).

10. In the Joins area for the new definition, select the appropriate fact column, then select the appropriate timestamp or numeric column from the Time dimension.

    After you create the joins, you can display the Hierarchies tab in the Time Dimension editor to view the default hierarchies and levels.

11. Edit the tables in the model. See **Editing Fact Tables and Dimension Tables**.

12. Click **Done** to return to the data model.

---

**Adding Measures and Attributes to a Data Model**

This topic describes how to add measures and attributes to your data model.
Video

Topics:
- Editing Measures and Attributes
- Specifying Aggregation for Measures in Fact Tables
- Creating Calculated Measures
- Creating Derived Attributes
- Creating Expressions in the Expression Editor
- Copying Measures and Attributes

Editing Measures and Attributes

Use the table editor to add, edit, and delete measures and attributes in your data model.

1. In Data Modeler, lock the model for editing.
2. Click the fact table or dimension table that contains the measure or attribute that you want to edit.
3. To edit all the columns directly in the table editor, select Edit All.
   To edit, copy, or delete a selection of columns at the same time, Shift + click or Ctrl + click the rows you want.

4. In the table editor, right-click a column and optionally click Copy or Delete as appropriate.
5. In the table editor, click the column that you want to edit or click Add Column.
6. Change settings on the Overview tab as needed.
   - Edit the display name and description.
   - Change the sort order.
     By default, columns are sorted based on the data in the column and reports display data in this order. To sort a column based on the data in another column, select Sort by a different column and select the Sort By value you prefer. For example, instead of sorting a Month Name attribute alphabetically, you could sort by month number, such as 1 (January), 2 (February), 3 (March), and so on.
7. Change settings for calculated measures or derived attributes, see Creating Calculated Measures and Creating Derived Attributes.

8. From the Permissions tab, optionally modify object permissions. See Securing Access to Objects in the Model.

9. From the Data Filters tab, optionally define data filters that provide row-level filtering for data model objects. See Defining Data Security Filters.

10. From the Levels tab for columns in a fact table, optionally create a level-based measure. See Setting Aggregation Levels for Measures.

11. Click Done to return to the table editor.

Specifying Aggregation for Measures in Fact Tables

You can specify aggregation for a measure in a fact table. For example, you can set the aggregation rule for a Revenue column to Sum.

See Setting Aggregation Levels for Measures.

1. In Data Modeler, lock the model for editing.

2. In the Fact Tables area, click the fact table for which you want to create measures.

3. In the Columns list, change the aggregation rule for the appropriate columns to specify that they're measures.

   To apply the same aggregation rule to multiple columns, Shift + click or Ctrl + click the appropriate columns.

   Aggregation options include:

   None: No aggregation.

   Sum: Calculates the sum by adding up all values.

   Average: Calculates the mean value.

   Median: Calculates the middle value.

   Count: Calculates the number of rows that aren't null.

   Count Distinct: Calculates the number of rows that aren't null. Each distinct occurrence of a row is counted only once.
Maximum: Calculates the highest numeric value.
Minimum: Calculates the lowest numeric value.
First: Selects the first occurrence of the item.
Last: Selects the last occurrence of the item.
Standard Deviation: Calculates the standard deviation to show the level of variation from the average.
Standard Deviation (all values): Calculates the standard deviation using the formula for population variance and standard deviation.

Tip:
Some calculated measures are Pre-Aggregated. These measures have calculations involving measures that already have an aggregation applied. To edit a calculation that contains pre-aggregated measures, click the column name. See also Creating Calculated Measures.

For most measures, the same aggregation rule applies for each dimension but for some measures you’ll want to specify one aggregation rule for a given dimension and specify other rules to apply to other dimensions.

Time dimensions are most likely to require different aggregation. For example, Headcount (calculated measure) typically aggregates as SUM across Organization and Geography dimensions but SUM does not apply for a Time dimension. Aggregation for the Time dimension should be LAST, so you can show Headcount on the last week or day of the year.

4. To override the aggregation for specific dimensions:
a. Click the name of the measure column.

b. Deselect **Same for all dimensions**.

c. Click **Add Override**.

d. Select the dimension you want to aggregate differently, for example Time.

e. Select an aggregation rule for the dimension.

f. If required, override aggregation for another dimension.

g. Click **Done**.

When dimension-specific aggregation rules are defined for a measure, you see an asterisk * next to the aggregation rule in the Columns table. For example, **Sum***.

5. By default, all the columns in the fact table are displayed in reports. Deselect the **Available** box for any columns that you don't want to display. You can use Shift + click or Ctrl + click to select multiple rows.

6. Click **Cancel** to cancel any of your changes.

7. Click **Done** to return to the table editor.

Creating Calculated Measures

If a fact table does not include all the measures that you need, then you can create calculated measures. For example, you can create a calculated measure called Average Order Size using the formula Revenue/Number of Orders.

1. In Data Modeler, lock the model for editing.

2. In the Fact Tables area, click the fact table for which you want to create measures.

3. In the Columns area, click **Add Column**.

4. In the New Column editor, enter a name and description for the column.

   Then, enter an expression directly in the Expression box, or click **Full Editor** to display the Expression editor.

   See **Creating Expressions in the Expression Editor**.

5. Expressions can contain measures that are already aggregated, as well as measures with no aggregation applied. Do one of the following:
- Set Aggregation to **Before Calculating**, if your expression includes measures that are already aggregated or aggregation is not required.

- Set Aggregation to **After Calculating** and select an aggregation rule, such as **Sum**, **Average**, **Count**, to apply aggregation after calculating the expression. See **Setting Aggregation Levels for Measures**.

For more information and examples, see **About Creating Calculated Measures**.

6. Click **Done** to return to the table editor.

---

### About Creating Calculated Measures

Calculated measures, as the name suggests, are calculated from other measures. For example, you can create a measure that calculates Average Order Size using the formula Revenue/Number of Orders.

Calculations can contain measures that are already aggregated, as well as measures with no aggregation applied. For example:

- **Calculation includes aggregated measures**: `Sum(Revenue)/Sum(Orders)`
- **Calculation includes measures with no aggregation applied**: `UnitPrice X Quantity`

If the measures in your calculation aren’t pre-aggregated, such as `UnitPrice` and `Quantity`, you may apply aggregation after the calculation. For example, `Sum(UnitPrice X Quantity)`.

Check the measures in your calculations before choosing whether to apply aggregation **Before Calculating** or **After Calculating** your expression. See **Creating Calculated Measures**.

### Calculations Include Measures Already Aggregated

Set Aggregation to **Before calculating** if the calculation contains pre-aggregated measures. For example: `Sum(Revenue)/Sum(Orders)`
Note:
If you select **After calculating**, any aggregation applied to measures in the calculation is ignored.

Calculations Include Non Aggregated Measures

Optionally, you can apply aggregation after your calculation. Set Aggregation to **After calculating** and then select an aggregation rule from the list. For example, **Sum**, **Average**, **Count** and so on.
Creating Derived Attributes

You can create custom or derived attributes for dimension tables that are based on an expression. For example, you can use an expression to concatenate multiple address columns into a single Full Address column.

1. In Data Modeler, lock the model for editing.
2. In the Dimension Tables area, click the dimension table for which you want to create derived attributes.
3. In the Columns area, click Add Column.
4. In the New Column editor, enter a name and description for the column. Then, enter an expression directly in the Expression box, or click Full Editor to display the Expression editor. See Creating Expressions in the Expression Editor.

You can use a variable in a column expression. See Defining Variables.

5. Click Done to return to the table editor.

Creating Expressions in the Expression Editor

You can use the Expression Editor to create constraints, aggregations, and other transformations on columns.

Topics:

• About the Expression Editor
Creating an Expression

About the Expression Editor

When modeling data, you can use the Expression Editor to create constraints, aggregations, and other transformations on columns. For example, you can use the Expression Editor to change the data type of a column from date to character. You can also use the Expression Editor to create expressions for data filters.

The Expression Editor contains the following sections:

- The Expression box on the left-hand side enables you to edit the current expression.
- The toolbar at the bottom contains commonly used expression operators, such as a plus sign, equals sign, or comma to separate items.
- The Expression Elements section on the right-hand side provides building blocks that you can use in your expression. Examples of elements are tables, columns, functions, and types.

The Expression Elements section only includes items that are relevant for your task. For example, if you open the Expression Editor to define a calculated measure, the Expression Elements section only includes the current fact table, any dimension tables joined to that table, plus any fact tables indirectly joined through a dimension table. Similarly, when you define a derived attribute, you see the current dimension table, any fact tables joined to that table, and any dimension table joined to those fact tables.

Another example is that time hierarchies are only included if the Time fact table is joined to the current table.

See Expression Editor Reference.
Creating an Expression

You can use the Expression Editor to create constraints, aggregations, and other transformations on columns.

1. Add or edit a column from the Table editor. See Editing Columns.
2. Enter an expression in the Expression box and click Done. Or, click Full Editor to launch the Expression Editor.
3. Use the Expression Elements menus to locate the building blocks you want to use to create your expression.

   Drag and drop an element to add it to your expression. You can also double-click an element to insert it, or you can select the element and click the arrow icon.

   When you add a function, brackets indicate text that needs to be replaced. Select the text, then type, or use the Expression Elements menus to add the appropriate element.

   See Expression Editor Reference.
4. Click Filter and then enter text in the search box to filter the available elements. Remove the text to revert to the full list of elements.
5. Click Actions to show or hide menus under Expression Elements, or to expand or collapse all menus.
6. Click an item on the toolbar to insert an operator.
7. Click Undo or Redo as needed when building your expression.
8. Click Validate to check your work.
9. Click **Save** when you’re finished.

### Copying Measures and Attributes

You can copy measures and attributes in your data model.

- From the Data Model menu in the left pane, right-click the column that you want to copy and select **Copy**.
  
  To copy multiple columns, Shift + click or Ctrl + click all the rows that you want and right-click to select **Copy**.

- From the Data Model menu in the left pane, click **Column Actions** for the column that you want to copy and select **Copy**.

  The copy is displayed with a number added to the name.

### Copying Model Objects

Sometimes it’s quicker to copy objects rather than starting from scratch.

In Data Modeler you can copy fact tables, dimension tables, database tables, and database views:

- **Fact tables**
  
  To copy an existing fact table, select **Duplicate** from the **Fact Table Actions** menu. When you copy a fact table, Data Modeler includes joins by default. See [Creating Fact Tables Individually](#).

  Aggregation level settings for measures aren't copied as, in most cases, level settings in the original fact table and the copied version differ. After copying a fact table, review and set the aggregation levels for measures as required.

- **Dimension tables**
  
  To copy an existing dimension table, select **Duplicate** from the **Dimension Table Actions** menu. When you copy a dimension table, Data Modeler excludes joins by default. See [Creating Dimension Tables Individually](#).

- **Database tables** and **views**
  
  To copy an existing database object, select **Duplicate** from the **Actions** menu. When you copy a table or view, Data Modeler creates a view based on the table or view you copy. See [Defining Source Views](#).
12

Defining Hierarchies and Levels for Drilling and Aggregation

You can define hierarchies and levels in Data Modeler.

Topics:
• Typical Workflow for Defining Hierarchies and Levels
• About Hierarchies and Levels
• Editing Hierarchies and Levels
• Setting Aggregation Levels for Measures

Typical Workflow for Defining Hierarchies and Levels

Here are the common tasks to add hierarchies and levels to your data model.

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>More Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add hierarchies and levels</td>
<td>Create hierarchies and levels for your dimension tables</td>
<td>Editing Hierarchies and Levels</td>
</tr>
<tr>
<td>Set aggregation levels for measures</td>
<td>Set custom aggregation levels for measures that are different from the default level</td>
<td>Setting Aggregation Levels for Measures</td>
</tr>
</tbody>
</table>

About Hierarchies and Levels

A hierarchy shows relationships among groups of columns in a dimension table. For example, quarters contain months and months contain days. Hierarchies enable drilling in reports.

A dimension table can have one or more hierarchies. A hierarchy typically begins with a total level, then has child levels, working down to the lowest detail level.

All hierarchies for a given dimension must have a common lowest level. For example, a time dimension might contain a fiscal hierarchy and a calendar hierarchy, with Day as the common lowest level. Day has two named parent levels called Fiscal Year and Calendar Year, which are both children of the All root level.

All levels, except the total level, must have at least one column specified as the key or display column. However, it’s not necessary to explicitly associate all of the columns from a table with levels. Any column that you don’t associate with a level is automatically associated with the lowest level in the hierarchy that corresponds to that dimension table.

There’s no limit to the number of levels you can have in a hierarchy. The total number of levels isn’t by itself a determining factor in query performance. However, be aware that for extremely complex queries, even a few levels can impact performance.
Editing Hierarchies and Levels

After creating dimension tables, you can add hierarchies and levels to those tables.

A hierarchy is a system of levels in a dimension that are related to each other by one-to-many relationships. For example, the Geography hierarchy might include levels for Country, State, and City.

When fact tables and dimension tables are joined, a default hierarchy is created. You can't add hierarchies for a particular dimension table until it has been joined to a fact table. Columns used in a join from a dimension table are used as key columns for the detail level in a hierarchy.

1. In Data Modeler, lock the model for editing.
2. In the Dimension Tables area, click the dimension table for which you want to add a hierarchy. The dimension table must have at least one join to a fact table.
3. In the Dimension editor, click the Hierarchies tab.
4. In the Hierarchies area, click Add Level. The upper part of the Add Level box shows dimension columns that haven't yet been used in a level.

   The lower part of the box shows shared levels that have already been used in another hierarchy for this table.

   Select the dimension column or shared level that you want to use.

5. Continue to click Add Level and select the appropriate columns or shared levels, until you've added all the levels.
6. Drag and drop levels to a different location in the order, as appropriate. You can also right-click a level and select Move left or Move right.

7. Click a level to display a dialog in which you can specify the level name, the key column, and the display column for the level.
When you edit a shared level, the changes are made in all instances of the level. For example, if you change Country Name to Country, the level name is changed in all hierarchies where the shared level is used.

8. To delete a level, right-click the level and then click **Delete**, or click the level and select **Delete level** from the Level dialog. You can't delete the default Total and Detail levels.

When you delete a shared level, you can choose whether to delete it from the current hierarchy only, or to delete it from all hierarchies.

For example, assume you want to delete the shared level Country Name from the current hierarchy only, called Hierarchy 1. Right-click the level in Hierarchy 1, select **Delete**, then select **from Hierarchy 1**.

**Tip:**

Selecting **Delete level** from the Level dialog only deletes the level from the current hierarchy. To remove a shared level from all hierarchies, right-click the level and select **Delete**, then select **from all hierarchies**.

9. Deselect **Available** if you don't want the hierarchy visible in analyses.

10. To add another hierarchy, click **Add Hierarchy** and repeat the steps in this procedure. Or, click **Done** when you're finished.

### Setting Dimension Table Properties for Hierarchies

From the Overview tab for a particular dimension table, you can set properties that apply to all hierarchies for that table.

1. In Data Modeler, lock the model for editing.
2. Click the dimension table that you want to edit.
3. Change settings on the Overview tab as needed:
   - **Time dimension** — Specifies that hierarchies for this dimension table support a time dimension. Hierarchies for time dimensions cannot include skip levels or be unbalanced.
   - **Enable skipped levels** — Specifies that this dimension table supports hierarchies with skipped levels. A skip-level hierarchy is a hierarchy where there are members that do not have a value for a particular ancestor level. For example, in a Country-State-City-District hierarchy, the city "Washington, D.C." does not belong to a State. In this case, you can drill down from the Country level (USA) to the City level (Washington, D.C.) and below.
   - **Enable unbalanced hierarchies** — Specifies that this dimension table supports unbalanced hierarchies. An unbalanced (or ragged) hierarchy is a hierarchy where the leaves (members with no children) don't necessarily have the same depth. For example, a site can choose to have data for the current month at the day level, previous months data at the month level, and the previous 5 years data at the quarter level.
Setting Aggregation Levels for Measures

You can set custom aggregation levels for a measure.

See About Setting Aggregation Levels for Measures.

1. In Data Modeler, lock the model for editing.

2. In the Fact Tables area, click the fact table in which the measure is located.

   Tip:
   The fact table must be joined to at least one dimension table.

3. Specify the aggregation rule for the new column that you want to become the level-based measure.

4. Click the column name, then click Levels.

5. In the Levels tab, for one or more hierarchies, use the slider to select the aggregation level for the measure.

6. Click Done to return to the table editor.

About Setting Aggregation Levels for Measures

By default, measures are aggregated at the level of the dimension attributes that are selected in an analysis. For example, in an analysis that includes Sales Person and Revenue columns, the Revenue is aggregated at the level of a Sales Person.

To calculate ratios, you often need measures that are aggregated at a level that is different than the grain of the analysis. For example, to calculate the Revenue Percent Contribution for a Sales Person with respect to his department, you need Department Revenue at the Sales Person level in an analysis (Sales Person, Revenue, Revenue *100 / Revenue@Dept). In this example, Revenue@Dept has a custom aggregation level that is different from the default level.
Securing Your Data Model

You can define object-level permissions and row-level security data filters for your data model.

Video

Topics:
• Typical Workflow for Securing Your Data
• Creating Variables to Use in Expressions
• Securing Access to Objects in the Model
• Securing Access to Data

Typical Workflow for Securing Model Data

Here are the common tasks to secure your data model.

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>More Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Define variables for data filters, if needed</td>
<td>Optionally, create variables that dynamically calculate and store values for use in column expressions and data filters.</td>
<td>Creating Variables to Use in Expressions</td>
</tr>
<tr>
<td>Set permissions on model objects</td>
<td>Object permissions control visibility for your entire model, or individual fact tables, dimension tables, and columns.</td>
<td>Securing Access to Objects in the Model</td>
</tr>
<tr>
<td>Define row-level security filters</td>
<td>Data filters limit results returned for fact tables, dimension tables, and columns.</td>
<td>Securing Access to Data</td>
</tr>
</tbody>
</table>

Creating Variables to Use in Expressions

In Data Modeler, you can define variables that dynamically calculate and store values so that you can use those values in column expressions or data filters.

Topics:
• About Variables
• Defining Variables

About Variables

Variables dynamically calculate and store values so that you can use those values in expressions. You can use variables in column expressions, or in data filters.
For example, suppose User1 belongs to Department1 and User2 belongs to Department2. Each user must access only the data that is specific to his department. You can use the DEPARTMENT_NUMBER variable to store the appropriate values for User1 and User2. You can use this variable in a data filter in which the data is filtered by Department2 for User1 and Department2 for User2. In other words, variables dynamically modify metadata content to adjust to a changing data environment.

Values in variables aren’t secure, because object permissions don’t apply to variables. Anybody who knows or can guess the name of the variable can use it in an expression. Because of this, don’t put sensitive data like passwords in variables.

You can’t use a variable in an expression that defines another variable.

**Defining Variables**

You can create a variable for use in column expressions and data filters. For example, a variable called SalesRegion might use a SQL query to retrieve the name of the sales region of the user.

---

**Tip:**

Only reference source database objects in the SQL query for a variable. Don’t include names of data model objects in the query.

---

1. In Data Modeler, lock the model for editing.
2. In the Variables menu in the left pane, click the **Plus** icon.
3. Enter a SQL query to populate the value of the variable:
   a. Specify whether the variable returns **A single value** or **Multiple values**.
   b. Enter a SQL query to populate the value or values of the variable. For example:
      - Return a single value with the query like: `SELECT prod-name FROM products`
      - Return multiple values with a query like: `SELECT 'MyVariable', prod-name FROM products`
      For multiple values, always use the format: `SELECT 'VariableName', VariableValue FROM Table`
   c. Provide a default starting value if needed.
   d. Click **Test** to validate that the query returns an appropriate value.
To create a variable that refreshes its value at the start of each user session, select **On sign in** for **Update Value**.

To create a variable that refreshes its value on a schedule that you set, select **On a schedule** for **Update Value**.

In the **Run SQL Query** area, select the frequency and start date for refreshing the variable.

To create a variable with a static value that never changes, select **Never** for **Update Value** and provide a value for the variable in the **Value** field.

Click **Done** to return to the data model.

**Tip:**

To edit an existing variable, right-click it in the Variables list and select **Inspect**. To delete a variable, right-click it and select **Delete**.

After you have defined a variable, you can use it in a data filter or in a column expression. See Defining Data Security Filters and Creating Derived Attributes.

### Securing Access to Objects in the Model

It's important to keep sensitive information secure. Everyone has access to the data in your model by default. To avoid exposing sensitive data, set show and hide permissions for your entire model or for individual fact tables, dimension tables, and columns.

For example, you can restrict access to certain Revenue columns to ensure only authorized users can view them. Or you can restrict access to an entire model to stop people opening the model or accesses its subject area.

1. In Data Modeler, lock the model for editing.
2. To restrict access to the whole model, select the **Permissions** tab.
To restrict access to a specific item in the model, edit the fact table, dimension table, or column whose access you want to secure, then select the **Permissions** tab.

3. To control access, click **Add** and select the appropriate role. Alternatively, in the left pane, click **Roles**. Then, drag and drop a role to the Permissions list. To add multiple roles, use Shift + click or Ctrl + click to make your selections before you drag and drop.

![Roles and Permissions](image)

4. Specify whether or not this object is visible to users with that role by selecting either **Visible** or **Hidden**.
   - Models — If you hide a model, users with that role can't open the model or its subject area.
   - Model objects — If you hide a fact table, dimension table, or column, users with that role can't see the object in reports.

   The same users will see the object in Data Modeler if they have the BI Data Model Author role and have access to the model.

5. To remove roles from the Permissions list, do one of the following:
   - Right-click a role and select **Remove**.
   - Select **Remove** from the Actions menu for that role.
   - Select multiple roles using Shift + click or Ctrl + click, then select **Remove Selected** from the Permissions Action menu.
   - Remove all roles by selecting **Remove All** from the Permissions Action menu.

   ![Remove](image)

   **Note:**
   You can’t remove the Everyone role.

### About Permission Inheritance

When multiple application roles act on a user or role with conflicting security attributes, the user or role is granted the least restrictive security attribute. Also, any explicit
permissions acting on a user take precedence over any permissions on the same objects granted to that user through application roles.

Tip:
If you deny access to a table, access to all columns in that table is implicitly denied as well.

Securing Access to Data

You can define data filters for fact tables, dimension tables, and columns that provide row-level security for data model objects. For example, you can create a filter that restricts access to the Products table so that only certain brands are visible to users assigned to a particular role.

1. In Data Modeler, lock the model for editing.
2. Edit the fact table, dimension table, or column you want to secure.
3. Select the Data Filters tab.
4. Add a role to the Data Filters list by doing one of the following:
   - Click Add and select the appropriate role.
   - In the left pane, click Roles. Then, drag and drop a role to the Data Filters list.
5. Enter an expression to specify which data is accessible for that role. Either enter the expression directly, or click Full Editor to display the Expression Editor.
   See Creating Expressions in the Expression Editor.
   You can use a variable in a data filter expression. See Defining Variables.
6. Select Enable to specify whether the filter is enabled for that role.
7. To remove filters from the Data Filters list, do one of the following:
   - Right-click a filter and select Remove.
   - Select Remove from the Actions menu for that filter.
   - Select multiple filters using Shift-click or Ctrl-click, then select Remove Selected from the Data Filters Action menu.
• Remove all filters by selecting **Remove All** from the Data Filters Action menu.

8. Click **Done**.
Part IV
Managing Your Service

This part explains how to manage your service. The information is aimed at administrators whose primary job is to manage users and keep them productive. Administrators perform a long list of critical duties; they control user permissions and amend accounts, set up database connections for data modelers, manage data storage to avoid exceeding storage limits, keep regular backups so users don't risk losing their work, authorize access to external content by whitelisting safe domains, troubleshoot user queries, and so much more.

Chapters:

• Managing What Users Can See and Do
• Taking Snapshots and Restoring
• Performing Administration Tasks
Managing What Users Can See and Do

Administrators can manage what other users are allowed to see and do when working with data.

Topics:
- Typical Workflow for Managing What Users See and Do
- About Users and Roles
- About Application Roles
- Configuring What Users Can See and Do
- Functionality Enabled by Application Roles

Typical Workflow for Managing What Users See and Do

Here are the common tasks to start managing what users can see and do when working with Oracle BI Cloud Service.

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>More Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Understand application roles</td>
<td>Learn about the predefined application roles and what they allow users to do in Oracle BI Cloud Service.</td>
<td>About Application Roles</td>
</tr>
<tr>
<td>Assign application roles to users</td>
<td>Give your users access to different features by granting them application roles.</td>
<td>Assigning Application Roles to Users</td>
</tr>
<tr>
<td>Assign application roles to user roles</td>
<td>Grant access to users more quickly through roles. Give a group of users access in one go.</td>
<td>Assigning Application Roles to User Roles</td>
</tr>
<tr>
<td>Add members and actions to application roles</td>
<td>Grant access to Oracle BI Cloud Service features in a different way. Go to the application role and assign users and groups from there.</td>
<td>Adding Members to Application Roles</td>
</tr>
<tr>
<td>Add your own application roles</td>
<td>Oracle BI Cloud Service provides application roles that map directly to all the main features but you can create your own application roles that make sense to your business too.</td>
<td>Adding Your Own Application Roles</td>
</tr>
</tbody>
</table>

About Users and Roles

Administrators manage users and roles through My Services and Oracle BI Cloud Service. Most administrators initially use My Services to set up user accounts and give people access to Oracle BI Cloud Service through roles. In the Oracle BI Cloud
Service Console, administrators see all the users and roles configured through My Services, plus they can fine tune user permissions through application roles.

**My Services**

The identity domain controls the authentication and authorization of users who sign in to Oracle Cloud services. When Oracle Cloud services are provisioned in an identity domain, several predefined roles and user accounts are available through My Services to help you get started. You can give people access to Oracle BI Cloud Service through these predefined roles.

<table>
<thead>
<tr>
<th>Predefined Roles (My Services)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Identity Domain Administrator</td>
<td>TenantAdminGroup</td>
</tr>
<tr>
<td>BI_SE BI Service Based Entitlement Administrator</td>
<td>BI_SE.BI_ServiceEntitlementAdministrator</td>
</tr>
<tr>
<td>Users in the organization that manage users and roles for an identity domain.</td>
<td></td>
</tr>
<tr>
<td>BIServiceName.BICloudServiceAdministrators</td>
<td>Users in the organization that administer Oracle BI Cloud Service.</td>
</tr>
<tr>
<td>BIServiceName.BICloudServiceAdvancedContentAuthors</td>
<td>Users in the organization that create content and model data in Oracle BI Cloud Service.</td>
</tr>
<tr>
<td>BIServiceName.BICloudServiceUsers</td>
<td>Users in the organization that view reports and explore data in Oracle BI Cloud Service.</td>
</tr>
<tr>
<td>DBServiceName.Database Administrator</td>
<td>Users within the organization that administer the database available with Oracle BI Cloud Service.</td>
</tr>
<tr>
<td>DBServiceName.Database Developer</td>
<td>Users within the organization that have the role of database developer for the database available with Oracle BI Cloud Service.</td>
</tr>
<tr>
<td>DBServiceName.Database User</td>
<td>Users within the organization that have the role of database user for the database available with Oracle BI Cloud Service.</td>
</tr>
</tbody>
</table>

See Adding Users and Assigning Roles and Oracle Cloud User Roles and Privileges in Managing and Monitoring Oracle Cloud.

**Oracle BI Cloud Service Console**

From the Console, administrators can see all the users and roles provisioned for the identity domain and give them appropriate permissions through application roles.

**About Application Roles**

An application role comprises a set of privileges that determine what users can see and do after signing in to Oracle BI Cloud Service. It’s your job as an administrator to assign people to one or more application roles.

There are two types of application role:

<table>
<thead>
<tr>
<th>Type of Application Role</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Predefined</td>
<td>Include a fixed set of privileges.</td>
</tr>
</tbody>
</table>
Predefined Application Roles

Oracle BI Cloud Service provides several predefined application roles to get you started. In many cases, these predefined application roles are all that you need.

Tip:

You can also create your own application roles. See Adding Your Own Application Roles.

<table>
<thead>
<tr>
<th>Predefined Application Role</th>
<th>Description</th>
<th>Default Members</th>
</tr>
</thead>
<tbody>
<tr>
<td>BI Service Administrator</td>
<td>Allows users to administer Oracle BI Cloud Service and delegate privileges to others.</td>
<td>Identity Domain Administrator</td>
</tr>
<tr>
<td>BI Data Model Author</td>
<td>Allows users to manage the data model in Oracle BI Cloud Service.</td>
<td>BI Service Administrator</td>
</tr>
<tr>
<td>BI Data Load Author</td>
<td>Allows users to load data using Oracle BI Cloud Service REST API and Oracle BI Cloud Service Data Sync.</td>
<td>BI Service Administrator</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Allows users to create analyses and dashboards in Oracle BI Cloud Service.</td>
<td>BI Data Model Author BI Discovery Content Author</td>
</tr>
<tr>
<td>BI Advanced Content Author</td>
<td>Allows users to perform more advanced content management tasks, such as add data sources for analyses and dashboards, and export dashboards.</td>
<td>BI Service Administrator</td>
</tr>
<tr>
<td>BI Discovery Content Author</td>
<td>Allows users to create visualization projects, explore data using Visual Analyzer, and add data sources for visualizations.</td>
<td>BI Advanced Content Author</td>
</tr>
<tr>
<td>BI Consumer</td>
<td>Allows users to view and run reports in Oracle BI Cloud Service (projects, analyses, dashboards). Use this application role to control who has access to the service.</td>
<td>BI Content Author</td>
</tr>
</tbody>
</table>

You can’t delete predefined application roles or remove default memberships.
Application roles can have users, roles, or other application roles as members. This means that a user who is a member of one application role might indirectly be a member of other application roles.

For example, a member of the BI Service Administrator application role inherits membership of other application roles, such as BI Data Model Author and BI Consumer. Any user that is a member of BI Service Administrator can do and see everything that these other application roles can do and see. Therefore you don't need to add a new user (for example, John) to all these application roles. You can simply add the user to the BI Service Administrator application role.

Application Role Hierarchy

This diagram illustrates the application role hierarchy in Oracle BI Cloud Service. It also shows you how predefined user roles assigned through My Services map to the application roles.

Why Is the Administrator Application Role Important?

You need the BI Service Administrator application role to access administrative options in the Console.

There must always be at least one person in your organization with the BI Service Administrator application role. This ensures there is always someone who can delegate permissions to others. If you remove yourself from the BI Service Administrator role you'll see a warning message. Consider adding yourself back to the this application role before you sign out. After you sign out, you won't be allowed to manage permissions through the Console to reinstate yourself.

No Users With the BI Service Administrator Application Role?

If no one has administrative privileges, ask your identity domain administrator to add you or another user to the `<serviceInstanceName>.BICloudServiceAdministrator role through My Services security pages. This role is a member of the Administrator application role and enables access to the user management pages in the Console.
Configuring What Users Can See and Do

Administrators assign application roles to determine what other users can see and do in Oracle BI Cloud Service.

Video

Topics:

• Getting Started with Application Roles
• Assigning Application Roles to Users
• Assigning Application Roles to Multiple Users Through Roles
• Adding Members to Application Roles
• Adding Your Own Application Roles
• Deleting Application Roles

Getting Started with Application Roles

Administrators configure what users see and do in Oracle BI Cloud Service from the Users and Roles Console page. This page presents user information in 3 different views:

<table>
<thead>
<tr>
<th>Users and Roles Page</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Users tab</td>
<td>Shows users from the identity domain associated with your service. You can't add or remove user accounts through the Users tab but you can assign users one or more application roles in Oracle BI Cloud Service.</td>
</tr>
<tr>
<td>Roles tab</td>
<td>Shows roles from the identity domain associated with your service. You can't add or remove roles (groups of users) through the Roles tab but you can assign them to one or more application roles in Oracle BI Cloud Service. From the Roles tab you can also see who belongs to each role.</td>
</tr>
<tr>
<td>Application Roles tab</td>
<td>Shows predefined application roles for Oracle BI Cloud Service together with any custom application roles you define. From the Application Roles tab you can assign application roles to multiple users, roles, and other application roles. You can also create application roles of your own and assign privileges to them through other application roles.</td>
</tr>
</tbody>
</table>
Assigning Application Roles to Users

The Users page lists all the users who can sign in to Oracle BI Cloud Service. The list of names comes directly from the identity domain associated with your service. It’s the administrator’s job to assign users to appropriate application roles.

**Note:**

You can’t add user accounts to the identity domain through the Users page. Use My Services to manage user accounts for the identity domain.

1. Click **Console**.
2. Click **Users and Roles**.
3. Click the **Users** tab.

4. To show everyone, leave the **Search** field blank and click **Show Members: All**.
   To filter the list by name, enter all or part of a user name in the **Search** filter and press enter. The search is case-insensitive, and searches both name and display name.
5. To see what application roles are assigned to a user:
   a. Select the user.
   b. Click the action menu and select **Manage Application Roles**.

The user’s current application role assignments are displayed in the **Selected Application Roles** pane.
For example, this image shows a user called Ed Ferguson assigned with the Sales Analysts application role.

6. To assign additional application roles or remove current assignments:
   a. Show available application roles. Click **Search** to display all the application roles.

      Alternatively, filter the list by **Name** and click **Search**.

   b. Use the shuttle controls to move application roles between the **Available Application Roles** list and the **Selected Application Roles** list.

      To find out what actions each application role allows, see Functionality Enabled by Application Roles.

   c. Click **OK**.

Assigning Application Roles to Multiple Users Through Roles

The Roles page shows you all the roles that people signing in belong to in their identity domain. The list of roles comes directly from the identity domain associated with your service. It’s often quicker to assign privileges to multiple users through their predefined identity domain roles, than it is to assign privileges to users one by one.

**Note:**

You can’t add roles to the identity domain through the Roles page. Use My Services to manage user accounts and roles for your identity domain.

You can assign application roles from the Roles page. You can also see who belongs to each role.

1. Click **Console**.
2. Click **Users and Roles**.
3. Click the **Roles** tab.
4. Look in the **Members** area to see who belongs to each role:

   The number of users and roles that are members are displayed on the page. Click a number, such as 1 in this image, to see the members in more detail.

5. To display all available roles, leave the **Search** field blank and **Show Members**: **All**.

   To filter the list by name, enter all or part of a role name in the **Search** filter and press enter. The search is case-insensitive, and searches both name and display name.

   Alternatively, use the **Show Members** filter to list roles that are members of a particular application role or belong to another role.

6. To see the current application roles assignments:
   a. Select the role.
   b. Click the action menu and select **Manage Application Roles**.

   Current application role assignments display in the **Selected Application Roles** pane.

7. To assign additional application roles or remove them:
   a. Click **Search** to display all available application roles.
      
      Alternatively, enter all or part of an application role name and click **Search**.
   b. Use the shuttle controls to move application roles between the **Available Application Roles** list and the **Selected Application Roles** list.
      
      To find out what actions each application role allows, see Functionality Enabled by Application Roles.
   c. Click **OK**.

### Adding Members to Application Roles

Application roles determine what people are allowed to see and do in Oracle BI Cloud Service. It’s the administrator’s job to assign appropriate application roles to everyone using the service and to manage the privileges of each application role.

You can make individuals (users) and groups of users (roles) from your identity domain members of an application role. You can add other application roles as members too. See **About Application Roles**.

Remember:
• Members inherit the privileges of an application role.

• Application roles inherit privileges from their parent (application roles).

You select members for an application role or change parent privileges using the Console.

1. Click **Console**.
2. Click **Users and Roles**.
3. Click the **Application Roles** tab.
4. To display all available application roles, leave the **Search** field blank and **Show Members: All**.

   To filter the list by name, enter all or part of an application role name in the **Search** filter and press Enter. The search is case-insensitive, and searches both name and display name.

5. Look in the **Members** area to see who belongs to each application role:

   ![Application Role Management](image)

   The number of users, roles, and application roles that are members displays on the page. Click a number, such as 5 in this image, to see those members in more detail (either users, roles or application roles).

6. To add new members or remove members from an application role:
   a. Click **Members**.
   b. Select either users, roles, or application roles from the **Type** box and click **Search** to show the current members.
   c. Use the shuttle controls to move members between the **Available** and **All Selected** list.

   Some application roles aren't eligible to be members and these are grayed. For example, you can't select a parent application role to be a member.

   **Note:**

   Users marked ‘absent’ no longer have an account in your identity domain. To remove absent users, use the shuttle control to move the user from the **All selected users** list to the **Available users** list.

   d. Click **OK**.
7. To see whether an application role, such as Sales Analyst, inherits privileges from other application roles:
   a. Click the action menu.

   b. Select **Manage Application Roles**.

   Inherited privileges are displayed in the Selected Application Roles pane.

   ![Manage Application Roles](image)

   In this example, the Sales Analyst application role inherits privileges from BI Content Author and BI Advanced Content Author. When you assign someone the Sales Analyst application role, you authorize them to perform actions allowed by both these application roles. See Functionality Enabled by Application Roles.

8. To add or remove privileges:
   a. Click **Search** to display all available application roles.
      Alternatively, enter all or part of an application role name and click **Search**.

   b. Use the shuttle controls to move application roles between the **Available Application Roles** list and the **Selected Application Roles** list.

      You can’t select application roles that are grayed out. Application roles are grayed out so you can’t create a circular membership tree.

   c. Click **OK**.

**Adding Your Own Application Roles**

Oracle BI Cloud Service provides a set of predefined application roles. You can also create application roles of your own to suit your own requirements.

For example, you can create an application role that only allows a select group of people to view specific folders or projects.
1. Click **Console**.
2. Click **Users and Roles**.
3. Click the **Application Roles** tab.
4. Click **Add**.

5. Enter a name and describe the application role. Click **Save**.
   Initially, new application roles don’t have any members or privileges.

6. Add members to the application role:
   a. Click the action menu.
   b. Select **Manage Members**.
   c. Select the members (users, roles or application roles) that you want assigned to this application role and move them to the **Selected** pane on the right.
      For example, you might want an application role that restricts access to everyone in your organization, except sales managers. To do this, move anyone who is a sales manager, to the **Selected** pane.
   d. Click **OK**.
   See also **Adding Members to Application Roles**.

7. Optionally, add privileges to the new application role:
   a. Click the action menu.
   b. Select **Manage Application Roles**.
   c. Click **Search**.
   d. Move all the application roles you want this application role to inherit to the **Selected Application Roles** pane, and click **OK**.
Deleting Application Roles

You can delete application roles that you created but no longer need.

1. Click **Console**.
2. Click **Users and Roles**.
3. Click the **Application Roles** tab.
4. Navigate to the application role you want to delete.
5. Click the action menu for the application role you want to delete and select **Remove**.
6. Click **OK**.

Functionality Enabled by Application Roles

Application roles determine what you can see and do in Oracle BI Cloud Service. This topic describes what you’re allowed to see and do with the predefined application roles. Information is organized in two ways:
• Application role by functionality
• Functionality by application role

### Application Role by Functionality

<table>
<thead>
<tr>
<th>Feature</th>
<th>Functionality in Oracle BI Cloud Service</th>
<th>Application Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access</td>
<td>Access to Data Modeler</td>
<td>BI Data Model Author</td>
</tr>
<tr>
<td>Access</td>
<td>Access to Data Sync</td>
<td>BI Data Load Author</td>
</tr>
<tr>
<td>Access</td>
<td>Access to Dashboards</td>
<td>BI Consumer</td>
</tr>
<tr>
<td>Access</td>
<td>Access to Catalog</td>
<td>BI Consumer</td>
</tr>
<tr>
<td>Access</td>
<td>Access to Export</td>
<td>BI Advanced Content Author</td>
</tr>
<tr>
<td>Access</td>
<td>Access to Metadata Dictionary</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>Access</td>
<td>Access to Mobile</td>
<td>BI Consumer</td>
</tr>
<tr>
<td>Access</td>
<td>Add Data Sources</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>Access</td>
<td>Create or Edit Analyses</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>Access</td>
<td>Create or Edit Analyses with Accessibility Option Enabled</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>Actions</td>
<td>Create or Edit Navigate Actions</td>
<td>BI Advanced Content Author</td>
</tr>
<tr>
<td>Admin: Catalog</td>
<td>Change Permissions</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>Admin: Connections</td>
<td>Manage Database Connections</td>
<td>BI Service Administrator</td>
</tr>
<tr>
<td>Admin: General</td>
<td>Configure SMTP Mail Server</td>
<td>BI Service Administrator</td>
</tr>
<tr>
<td>Admin: General</td>
<td>Configure Virus Scanner</td>
<td>BI Service Administrator</td>
</tr>
<tr>
<td>Admin: General</td>
<td>Manage Session Information</td>
<td>BI Service Administrator</td>
</tr>
<tr>
<td>Admin: General</td>
<td>Manage Map Data</td>
<td>BI Service Administrator</td>
</tr>
<tr>
<td>Admin: General</td>
<td>Manage Dashboards</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>Admin: Security</td>
<td>Manage Application Roles</td>
<td>BI Service Administrator</td>
</tr>
<tr>
<td>Admin: Security</td>
<td>Set Ownership of Catalog Objects</td>
<td>BI Service Administrator</td>
</tr>
<tr>
<td>Admin: Security</td>
<td>Access to Application Role Management</td>
<td>BI Consumer</td>
</tr>
<tr>
<td>Admin: Security</td>
<td>Users/Roles - Can View Users and Roles</td>
<td>BI Consumer</td>
</tr>
<tr>
<td>Admin: Security</td>
<td>Application Roles - Can View Application Roles</td>
<td>BI Consumer</td>
</tr>
<tr>
<td>Admin: Security</td>
<td>Manage Safe Domains</td>
<td>BI Service Administrator</td>
</tr>
<tr>
<td>Admin: Search</td>
<td>Manage Search Indexing</td>
<td>BI Service Administrator</td>
</tr>
<tr>
<td>Admin: Snapshots</td>
<td>Back up and Restore with Snapshots</td>
<td>BI Service Administrator</td>
</tr>
<tr>
<td>Admin: Snapshots</td>
<td>Upload Data Model from a File (.rpd)</td>
<td>BI Service Administrator</td>
</tr>
<tr>
<td>Catalog</td>
<td>Personal Storage —My Folders and My Dashboard</td>
<td>BI Consumer</td>
</tr>
<tr>
<td>Catalog</td>
<td>Reload Server Metadata</td>
<td>BI Service Administrator</td>
</tr>
<tr>
<td>Catalog</td>
<td>BI Data Model Author</td>
<td></td>
</tr>
<tr>
<td>Catalog</td>
<td>See Hidden Items</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>Catalog</td>
<td>Create Folders</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>Feature</td>
<td>Functionality in Oracle BI Cloud Service</td>
<td>Application Role</td>
</tr>
<tr>
<td>------------------</td>
<td>------------------------------------------</td>
<td>---------------------------------</td>
</tr>
<tr>
<td>Catalog</td>
<td>Archive Catalog</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>Catalog</td>
<td>Unarchive Catalog</td>
<td>BI Service Administrator</td>
</tr>
<tr>
<td>Catalog</td>
<td>Perform Global Search</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>Conditions</td>
<td>Create Conditions</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>Dashboards</td>
<td>Save Customizations</td>
<td>BI Consumer</td>
</tr>
<tr>
<td>Dashboards</td>
<td>Assign Default Customizations</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>Dashboards</td>
<td>Create Bookmark Links</td>
<td>BI Consumer</td>
</tr>
<tr>
<td>Dashboards</td>
<td>Export Entire Dashboard to Excel</td>
<td>BI Advanced Content Author</td>
</tr>
<tr>
<td>Dashboards</td>
<td>Export Single Dashboard Page to Excel</td>
<td>BI Advanced Content Author</td>
</tr>
<tr>
<td>Home</td>
<td>Access Home Page</td>
<td>BI Consumer</td>
</tr>
<tr>
<td>Home</td>
<td>Access to Search</td>
<td>BI Consumer</td>
</tr>
<tr>
<td>Home</td>
<td>Access to Recent Activity</td>
<td>BI Consumer</td>
</tr>
<tr>
<td>Home</td>
<td>Create Menu</td>
<td>BI Consumer</td>
</tr>
<tr>
<td>Home</td>
<td>Dashboards Menu</td>
<td>BI Consumer</td>
</tr>
<tr>
<td>Home</td>
<td>Favorites</td>
<td>BI Consumer</td>
</tr>
<tr>
<td>Home</td>
<td>My Account Link</td>
<td>BI Consumer</td>
</tr>
<tr>
<td>My Account</td>
<td>Access to My Account</td>
<td>BI Consumer</td>
</tr>
<tr>
<td>My Account</td>
<td>Change Preferences</td>
<td>BI Consumer</td>
</tr>
<tr>
<td>Analysis</td>
<td>Add Data Source</td>
<td>BI Advanced Content Author</td>
</tr>
<tr>
<td>Analysis</td>
<td>Create Views</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>Analysis</td>
<td>Create Prompts</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>Analysis</td>
<td>Edit Column Formulas</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>Analysis</td>
<td>Edit Column Formulas</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>Analysis</td>
<td>Create Advanced Filters and Set Operations</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>Answers</td>
<td>Save Filters</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>Mobile</td>
<td>Enable Local Content</td>
<td>BI Consumer</td>
</tr>
<tr>
<td>Mobile</td>
<td>Enable Search</td>
<td>BI Consumer</td>
</tr>
<tr>
<td>Subject Area</td>
<td>Create and Edit Analyses</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>View Column Selector View</td>
<td>Add/Edit Column Selector View</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>View Compound Layout View</td>
<td>Add/Edit Compound Layout View</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>View Graph</td>
<td>Add/Edit GraphView</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>View Funnel</td>
<td>Add/Edit FunnelView</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>View Gauge</td>
<td>Add/Edit GaugeView</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>View Heat Matrix</td>
<td>Add/Edit Heat MatrixView</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>View Map</td>
<td>Add/Edit MapView</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>View Micro Chart</td>
<td>Add/Edit Micro Chart View</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>Feature</td>
<td>Functionality in Oracle BI Cloud Service</td>
<td>Application Role</td>
</tr>
<tr>
<td>----------------------</td>
<td>-----------------------------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>View Filters</td>
<td>Add/Edit FiltersView</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>View Dashboard Prompt</td>
<td>Add/Edit Dashboard PromptView</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>View Performance Tile</td>
<td>Add/Edit Performance TileView</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>View Static Text</td>
<td>Add/Edit Static TextView</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>View Narrative</td>
<td>Add/Edit NarrativeView</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>View No Results</td>
<td>Add/Edit No ResultsView</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>View Pivot Table</td>
<td>Add/Edit Pivot TableView</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>View Report Prompt</td>
<td>Add/Edit Report PromptView</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>View Selection Steps</td>
<td>Add/Edit Selection StepsView</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>View Logical SQL</td>
<td>Add/Edit Logical SQLView</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>View Table</td>
<td>Add/Edit TableView</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>View Title</td>
<td>Add/Edit TitleView</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>View Treemap</td>
<td>Add/Edit TreemapView</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>View Trellis</td>
<td>Add/Edit TrellisView</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>View View Selector</td>
<td>Add/Edit View SelectorView</td>
<td>BI Content Author</td>
</tr>
<tr>
<td>Data Visualization Projects</td>
<td>View and Explore Visualizations</td>
<td>BI Consumer</td>
</tr>
<tr>
<td>Data Visualization Projects</td>
<td>Create or Edit Data Visualization Projects</td>
<td>BI Discovery Content Author</td>
</tr>
<tr>
<td>Data Visualization Projects</td>
<td>Add Data Sources for Data Visualization Projects</td>
<td>BI Discovery Content Author</td>
</tr>
<tr>
<td>Data Visualization Projects</td>
<td>Manage Your Own Data File Uploads</td>
<td>BI Discovery Content Author</td>
</tr>
<tr>
<td>Data Visualization Projects</td>
<td>Manage All Data File Uploads</td>
<td>BI Service Administrator</td>
</tr>
</tbody>
</table>

### Functionality by Application Role

<table>
<thead>
<tr>
<th>Application Role</th>
<th>Functionality in Oracle BI Cloud Service</th>
<th>Feature</th>
</tr>
</thead>
<tbody>
<tr>
<td>BI Consumer</td>
<td>Access to Dashboards</td>
<td>Access</td>
</tr>
<tr>
<td>BI Consumer</td>
<td>Access to Mobile</td>
<td>Access</td>
</tr>
<tr>
<td>Access to Export</td>
<td>Access</td>
<td></td>
</tr>
<tr>
<td>BI Consumer</td>
<td>Personal Storage - My Folders and My Dashboard</td>
<td>Catalog</td>
</tr>
<tr>
<td>BI Consumer</td>
<td>Save Customizations</td>
<td>Dashboards</td>
</tr>
<tr>
<td>BI Consumer</td>
<td>Create Bookmark Links</td>
<td>Dashboards</td>
</tr>
<tr>
<td>Application Role</td>
<td>Functionality in Oracle BI Cloud Service</td>
<td>Feature</td>
</tr>
<tr>
<td>---------------------------</td>
<td>------------------------------------------------------------------------------------</td>
<td>----------------</td>
</tr>
<tr>
<td>BI Consumer</td>
<td>Access Home Page</td>
<td>Home</td>
</tr>
<tr>
<td>BI Consumer</td>
<td>Access Catalog UI</td>
<td>Home</td>
</tr>
<tr>
<td>BI Consumer</td>
<td>Simple Search Field</td>
<td>Home</td>
</tr>
<tr>
<td>BI Consumer</td>
<td>Recent Menu</td>
<td>Home</td>
</tr>
<tr>
<td>BI Consumer</td>
<td>Create Menu</td>
<td>Home</td>
</tr>
<tr>
<td>Create Navigate Actions</td>
<td>Home</td>
<td></td>
</tr>
<tr>
<td>BI Consumer</td>
<td>Dashboards Menu</td>
<td>Home</td>
</tr>
<tr>
<td>BI Consumer</td>
<td>Favorites</td>
<td>Home</td>
</tr>
<tr>
<td>BI Consumer</td>
<td>My Account Link</td>
<td>Home</td>
</tr>
<tr>
<td>BI Consumer</td>
<td>Access to My Account</td>
<td>My Account</td>
</tr>
<tr>
<td>BI Consumer</td>
<td>Change Preferences</td>
<td>My Account</td>
</tr>
<tr>
<td>BI Consumer</td>
<td>Enable Local Content</td>
<td>Mobile</td>
</tr>
<tr>
<td>BI Consumer</td>
<td>Enable Search</td>
<td>Mobile</td>
</tr>
<tr>
<td>BI Consumer</td>
<td>View and Explore Data Visualization Projects</td>
<td>Data Visualization</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Create and Edit Analyses</td>
<td>Access</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Create or Edit Analyses with Accessibility Option Enabled</td>
<td>Access</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Access to Metadata Dictionary</td>
<td>Access</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Change Permissions</td>
<td>Admin: Catalog</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Manage Dashboards</td>
<td>Admin: General</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>See Hidden Items</td>
<td>Catalog</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Create Folders</td>
<td>Catalog</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Perform Global Search</td>
<td>Catalog</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Archive Catalog Folders and Items</td>
<td>Catalog</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Create Conditions</td>
<td>Conditions</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Assign Default Customizations</td>
<td>Dashboards</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Create Views</td>
<td>Analysis</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Create Prompts</td>
<td>Analysis</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Edit Column Formulas</td>
<td>Analysis</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Edit Column Formulas</td>
<td>Analysis</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Create Advanced Filters and Set Operations</td>
<td>Analysis</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Save Filters</td>
<td>Answers</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Create and Edit Analyses</td>
<td>Subject Area</td>
</tr>
<tr>
<td>Add Data Sources</td>
<td>Access</td>
<td></td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Add/Edit Column Selector View</td>
<td>View Column Selector</td>
</tr>
<tr>
<td>Application Role</td>
<td>Functionality in Oracle BI Cloud Service</td>
<td>Feature</td>
</tr>
<tr>
<td>-------------------------</td>
<td>-----------------------------------------</td>
<td>--------------------------------</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Add/Edit Compound LayoutView</td>
<td>View Compound Layout</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Add/Edit GraphView</td>
<td>View Graph</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Add/Edit FunnelView</td>
<td>View Funnel</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Add/Edit GaugeView</td>
<td>View Gauge</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Add/Edit Micro Chart View</td>
<td>View Micro Chart</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Add/Edit FiltersView</td>
<td>View Filters</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Add/Edit Dashboard PromptView</td>
<td>View Dashboard Prompt</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Add/Edit Performance TileView</td>
<td>View Performance Tile</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Add/Edit Static TextView</td>
<td>View Static Text</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Add/Edit NarrativeView</td>
<td>View Narrative</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Add/Edit No ResultsView</td>
<td>View No Results</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Add/Edit Pivot TableView</td>
<td>View Pivot Table</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Add/Edit Report PromptView</td>
<td>View Report Prompt</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Add/Edit Selection StepsView</td>
<td>View Selection Steps</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Add/Edit Logical SQLView</td>
<td>View Logical SQL</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Add/Edit TableView</td>
<td>View Table</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Add/Edit HeatMatrixView</td>
<td>View Heat Matrix</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Add/Edit MapView</td>
<td>View Map</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Add/Edit TitleView</td>
<td>View Title</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Add/Edit TreemapView</td>
<td>View Treemap</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Add/Edit TrellisView</td>
<td>View Trellis</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Add/Edit View SelectorView</td>
<td>View View Selector</td>
</tr>
<tr>
<td>BI Content Author</td>
<td>Add/Edit View SelectorView</td>
<td>View View Selector</td>
</tr>
<tr>
<td>BI Discovery Content Author</td>
<td>Create Data Visualization Projects and Explore Data</td>
<td>Data Visualization</td>
</tr>
<tr>
<td>BI Discovery Content Author</td>
<td>Add Data Sources for Data Visualization Projects</td>
<td>Data Visualization</td>
</tr>
<tr>
<td>BI Discovery Content Author</td>
<td>Manage Your Data Sources</td>
<td>Console</td>
</tr>
<tr>
<td>BI Advanced Content Author</td>
<td>Access to Export</td>
<td>Access</td>
</tr>
<tr>
<td>BI Advanced Content Author</td>
<td>Create Navigate Actions</td>
<td>Actions</td>
</tr>
<tr>
<td>BI Advanced Content Author</td>
<td>Add Data Sources</td>
<td>Access</td>
</tr>
<tr>
<td>BI Advanced Content Author</td>
<td>Export Entire Dashboard to Excel</td>
<td>Dashboards</td>
</tr>
<tr>
<td>BI Advanced Content Author</td>
<td>Export Single Dashboard Page to Excel</td>
<td>Dashboards</td>
</tr>
<tr>
<td>BI Data Load Author</td>
<td>Access to Data Sync</td>
<td>Access</td>
</tr>
<tr>
<td>Application Role</td>
<td>Functionality in Oracle BI Cloud Service</td>
<td>Feature</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>------------------------------------------------------------------------------------------------------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>BI Data Load Author</td>
<td>Access to Oracle BI Cloud Service REST API</td>
<td>Access</td>
</tr>
<tr>
<td>BI Data Model Author</td>
<td>Access to Data Modeler</td>
<td>Access</td>
</tr>
<tr>
<td>BI Data Model Author</td>
<td>Reload Server Metadata</td>
<td>Catalog</td>
</tr>
<tr>
<td>BI Service Administrator</td>
<td>Manage Database Connections</td>
<td>Admin: Connections</td>
</tr>
<tr>
<td>BI Service Administrator</td>
<td>Manage Session Information</td>
<td>Admin: General</td>
</tr>
<tr>
<td>BI Service Administrator</td>
<td>Manage Map Data</td>
<td>Admin: General</td>
</tr>
<tr>
<td>BI Service Administrator</td>
<td>Manage Data File Uploads</td>
<td>Admin: General</td>
</tr>
<tr>
<td>BI Service Administrator</td>
<td>Manage Application Roles</td>
<td>Admin: Security</td>
</tr>
<tr>
<td>BI Service Administrator</td>
<td>Configure Virus Scanner</td>
<td>Admin: Security</td>
</tr>
<tr>
<td>BI Service Administrator</td>
<td>Configure SMTP Mail Server</td>
<td>Admin: Security</td>
</tr>
<tr>
<td>BI Service Administrator</td>
<td>Manage Safe Domains</td>
<td>Admin: Security</td>
</tr>
<tr>
<td>BI Service Administrator</td>
<td>Set Ownership of Catalog Objects</td>
<td>Admin: Security</td>
</tr>
<tr>
<td>BI Service Administrator</td>
<td>Manage Search Indexing</td>
<td>Admin: Search</td>
</tr>
<tr>
<td>BI Service Administrator</td>
<td>Backup and Restore with Snapshots</td>
<td>Admin: Snapshots</td>
</tr>
<tr>
<td>BI Service Administrator</td>
<td>Upload a Data Model from a File (.rpd)</td>
<td>Admin: Snapshots</td>
</tr>
<tr>
<td>BI Service Administrator</td>
<td>Reload Server Metadata</td>
<td>Catalog</td>
</tr>
<tr>
<td>BI Service Administrator</td>
<td>Unarchive Catalog Archives</td>
<td>Catalog</td>
</tr>
</tbody>
</table>
Taking Snapshots and Restoring

This topic describes how to back up and restore application content using a file called a snapshot.

Video

Topics:
- Typical Workflow for Taking Snapshots and Restoring
- About Snapshots
- Taking Snapshots and Restoring Information
- Downloading, Uploading, and Migrating Snapshots

Typical Workflow for Taking Snapshots and Restoring

Here are the common tasks to back up and restore your content using snapshots.

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>More Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Take a snapshot</td>
<td>Capture the data model, catalog content, and application roles in Oracle BI Cloud Service at a point in time.</td>
<td>Taking a Snapshot</td>
</tr>
<tr>
<td>Restore from a snapshot</td>
<td>Restore the system to a previously working state.</td>
<td>Restoring from a Snapshot</td>
</tr>
<tr>
<td>Delete a snapshot</td>
<td>Delete unwanted snapshots.</td>
<td>Deleting Snapshots</td>
</tr>
<tr>
<td>Download a snapshot</td>
<td>Save a snapshot to a local file system.</td>
<td>Downloading Snapshots</td>
</tr>
<tr>
<td>Upload a snapshot</td>
<td>Upload content from a snapshot that is stored on a local file system.</td>
<td>Uploading Snapshots</td>
</tr>
<tr>
<td>Migrate snapshot data</td>
<td>Migrate content to another environment.</td>
<td>Migrating Snapshot Data</td>
</tr>
</tbody>
</table>

About Snapshots

A snapshot captures the state of your environment at a point in time. Snapshots don't include data that is hosted on external data sources.

Take a snapshot of your environment before people start using the system and again at suitable intervals so you can restore the environment if something goes wrong.

<table>
<thead>
<tr>
<th>Artifacts Saved In a Snapshot</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data model</td>
<td>A snapshot of the data model created using Data Modeler.</td>
</tr>
</tbody>
</table>
Taking Snapshots and Restoring Information

You can take a snapshot of your system at any time.

Topics:
- Taking a Snapshot
- Restoring from a Snapshot
- Editing Snapshot Descriptions
- Deleting Snapshots

Taking a Snapshot

Administrators can take a snapshot of the system at any time.

1. Click Console.
2. Click Snapshots and Models.
3. Click New Snapshot.
4. Enter a short description for the snapshot to help you remember later why you took it.
   For example, why you created the snapshot and what it contains.
5. Click OK.
The latest content is saved to a snapshot.

Restoring from a Snapshot

If something goes wrong, you can easily restore your content to a previous working state from a snapshot.

⚠️ Caution:

- Restoring from a snapshot overwrites all existing content.
- Everyone who is currently signed-in has their session terminated.
- Any content created since the last snapshot will be lost.
- Large snapshot files take some time to upload and restore.
- Restored content takes a few minutes to refresh through your system. For large snapshots, allow up to 15–20 minutes.

1. Click Console.
2. Click Snapshots and Models.
3. Select the snapshot that you want to use to restore your system.
4. Click Manage Snapshot.
5. Click Restore to return your system to the state when this snapshot was taken.
6. In the Restore Snapshot dialog, select only those elements you want to restore.
   For example, you may not want to include application roles if you're restoring a snapshot taken from a pre-production service, to a production service. Pre-production roles often have different members to the production service. If so, deselect Application Roles before clicking Restore.
7. For auditing purposes, enter the reason why you're restoring.

8. Click **Restore**.

A warning message is displayed because restoring a snapshot can be very disruptive.

9. Click **Yes** to restore the selected snapshot, or click **No** to abandon the restore.

The time it takes to restore your system depends on the size of your snapshot. After the restore completes, you might need to wait a few more minutes for the restored content to refresh through your system. Sign out, then sign back in after, say, 15 or 20 minutes to inherit newly restored application roles, if any.

**Editing Snapshot Descriptions**

You can add or update the description for any snapshot.

1. Click **Console**.

2. Click **Snapshots and Models**.

3. Select the snapshot you want to edit.

4. Click **Manage Snapshot**.

5. Click **Edit**.

6. Update the description, and click **OK**.

**Deleting Snapshots**

From time to time, delete snapshots that you don't need.

1. Click **Console**.

2. Click **Snapshots and Models**.
Select the snapshot that you want to delete.

4. Click Manage Snapshot.

5. Click Delete to delete the snapshot.
   A message displays at the top right hand side of the page. If you change your mind, click Undo.

Downloading, Uploading, and Migrating Snapshots

Download and upload features enable you to save snapshots to your local file system and upload them back to the cloud. Use these features to back up and restore your content or to migrate content between development, test, and production environments.

Topics:
- Downloading Snapshots
- Uploading Snapshots
- Migrating Snapshot Data

Downloading Snapshots

Use the Download option to save a snapshot to your local file system. This allows you to locally store and manage snapshots you take of your system.

If you haven't taken the snapshot yet, you'll need to do that first. See Taking a Snapshot.

1. Click Console.
2. Click Snapshots and Models.
3. Select the snapshot that you want to download.
4. Click Manage Snapshot.
5. Click Download.

6. Enter and confirm a password for the snapshot. The password must contain at least 8 characters. Don't forget this password. You'll be asked for this password if you try to upload the file in the future. For example, you may want to restore or migrate the content stored in the snapshot.

7. Click OK.

What happens next depends on your browser. In Internet Explorer, for example, you browse the Save As dialog, and click Save to save the snapshot in the selected location. The snapshot downloads as an Oracle Business Intelligence archive file (.bar).

### Uploading Snapshots

You can upload a snapshot that you previously saved on your local file system.

When you upload a snapshot, the file itself is uploaded to your system but the artifacts stored inside the snapshot aren't immediately available in your environment. Any snapshot you upload displays in the snapshot list. When you're ready to do so, overwrite current artifacts, such as your catalog, with information from the snapshot. See Restoring from a Snapshot.

1. Click Console.
2. Click Snapshots and Models.
3. Click Upload Snapshot.

4. Use Browse to locate the snapshot that you want to upload.
   
   Select the Oracle Business Intelligence archive file (.bar) that contains your snapshot. You can only upload snapshots taken from Oracle BI Cloud Service.

5. Enter the snapshot password.
   
   You set the password during the download process.

6. Click OK.

The uploaded snapshot is displayed in the list of saved snapshots. To restore from a snapshot, see Restoring from a Snapshot.
Migrating Snapshot Data

You can migrate content users have created in one Oracle BI Cloud Service environment to another environment, using snapshots. For example, you may want to move pre-production content to a production environment.

1. Download the snapshot that you want to migrate to your local file system.
   See Downloading Snapshots.
2. Sign in to the target system and upload the snapshot.
   See Uploading Snapshots.
3. Select the newly uploaded snapshot in the list of saved snapshots.
   To migrate content, see Restoring from a Snapshot.
Performing Administration Tasks

This topic describes tasks performed by administrators managing Oracle BI Cloud Service.

Topics:

• Typical Workflow for Performing Administration Tasks
• Understanding Administration Tools
• Managing Database Connections
• Deleting Unused Data Sources
• Uploading Data Models from Oracle BI Enterprise Edition 12c
• Managing Map Information
• Whitelisting Safe Domains
• Managing How Content is Indexed and Searched
• Monitoring Users and Activity Logs
• Executing Test SQL Queries
• Restarting Your Service

Typical Workflow for Performing Administration Tasks

Here are the common tasks for administration.

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>More Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manage what users see and do</td>
<td>Configure what users see and do in Oracle BI Cloud Service from the</td>
<td>Managing What Users Can See and Do</td>
</tr>
<tr>
<td></td>
<td>Application Role page in the Console.</td>
<td></td>
</tr>
<tr>
<td>Back up and restore content</td>
<td>Back up and restore the data model, catalog content, and application roles</td>
<td>Taking Snapshots and Restoring</td>
</tr>
<tr>
<td></td>
<td>using a file called a snapshot.</td>
<td></td>
</tr>
<tr>
<td>Create database connections</td>
<td>Connect Oracle BI Cloud Service to one or more databases.</td>
<td>Managing Database Connections</td>
</tr>
<tr>
<td>Free up storage space</td>
<td>Delete data sources on behalf of other users to free up storage space.</td>
<td>Deleting Unused Data Sources</td>
</tr>
<tr>
<td>Manage how content is indexed and</td>
<td>Set up how content is indexed and crawled so users always find the latest</td>
<td>Managing How Content is Indexed and Searched</td>
</tr>
<tr>
<td>searched</td>
<td>information when they search.</td>
<td></td>
</tr>
<tr>
<td>Manage maps</td>
<td>Manage map layers and background maps.</td>
<td>Managing Map Information</td>
</tr>
<tr>
<td>Whitelist safe domains</td>
<td>Authorize access to safe domains.</td>
<td>Whitelisting Safe Domains</td>
</tr>
</tbody>
</table>
## Understanding Administration Tools

You administer Oracle BI Cloud Service from the Console and My Services (Oracle Cloud).

You must have the BI Service Administrator role to perform all the administration tasks outlined here.

<table>
<thead>
<tr>
<th>Product</th>
<th>Administration Tool</th>
<th>Description and How to Access</th>
</tr>
</thead>
</table>
| Oracle BI Cloud Service  | Console             | Use the Console to manage user permissions, back up your information, add database connections, and free up storage space for your service. See who is currently signed in, manage map information, whitelist safe domains, and diagnose issues with SQL queries:  
  • Managing What Users Can See and Do  
  • Taking Snapshots and Restoring  
  • Managing Database Connections  
  • Managing Map Information  
  • Deleting Unused Data Sources  
  • Whitelisting Safe Domains  
  • Monitoring Users and Activity Logs  
  • Executing Test SQL Queries |
| Oracle Cloud             | My Services         | Use My Services to manage user accounts and monitor usage metrics. |

## Managing Database Connections

Administrators create and manage cloud database connections for Oracle BI Cloud Service. Your business data doesn't have to be in one place. Connect to multiple cloud databases so business modelers and analysts can analyze company data wherever it is stored.

### Topics

- About Database Connections
- Connecting to Data in an Oracle Cloud Database

## About Database Connections

Oracle BI Cloud Service can handle data stored in Oracle Cloud databases. Simply connect Oracle BI Cloud Service to your cloud data sources to start analyzing the
data. It doesn’t matter if your business data is stored in several different locations. As you can connect Oracle BI Cloud Service to multiple cloud databases, business analysts can analyze all their data wherever it is stored.

Administrators create, manage, and test database connections for Oracle BI Cloud Service. Business modelers can see connected databases through Data Modeler and build business models from the data.

**About the Default Database Connection**

Your Oracle BI Cloud Service connects to Database Schema Service through the **Default Connection**. You can’t delete this connection. The Default Connection is always available.

Database Schema Service is the single schema-based service integrated with Oracle BI Cloud Service. You don’t have to create a connection to use this database.

**Databases You Can Connect To**

You can connect Oracle BI Cloud Service to Oracle Cloud databases. The target database must be Oracle Database Cloud Service.

**Database Connections for Data Models Uploaded from Oracle BI Enterprise Edition**

You don’t need to enter database connection information for data models pre-built with Oracle BI Enterprise Edition. Connection information for these models is in the data model file (.rpd) that you upload to Oracle BI Cloud Service. See About Uploading Oracle BI Enterprise Edition Data Models to the Cloud.

## Connecting to Data in an Oracle Cloud Database

Administrators create database connections so business analysts can analyze data stored in Oracle Cloud.

See About Database Connections.

1. Click **Console**.
2. Click **Connections**.
3. Click **New Connection**.
4. Enter a meaningful **Name** and **Description** that you’ll remember and business modelers will recognize.
5. Enter database connection information for your Oracle Database Cloud Service. Ask the database administrator to provide the connection details.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Connect Using</strong></td>
<td>Select how you want to connect to the database.</td>
</tr>
<tr>
<td><strong>Host</strong></td>
<td>Address of the database server or the host’s name.</td>
</tr>
<tr>
<td><strong>Port</strong></td>
<td>Port number on which the database server is listening for incoming connections.</td>
</tr>
<tr>
<td><strong>SID</strong> or <strong>Service Name</strong></td>
<td>SID — Name of the Oracle database instance. Service Name — Network service name of the database.</td>
</tr>
<tr>
<td>Option</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td><strong>Enable SSL</strong></td>
<td>Select to secure this connection using SSL. If you haven’t done so already, upload a wallet with SSL certificates. See <a href="#">Securing Database Connections with SSL</a>.</td>
</tr>
</tbody>
</table>
| **TNS Descriptor** | TNS connect descriptor that provides the location of the database and the name of the database service. You must select this option if you want to connect to an Oracle Real Application Cluster (RAC) database. Use the format:  
DESCRIPTION=(ADDRESS=(PROTOCOL=protocol)(HOST=host)(PORT=port))(CONNECT_DATA=(SERVICE_NAME=service name))  
For example:  
DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=myhost1.company.com)(PORT=1521))(CONNECT_DATA=(SERVICE_NAME=sales.company.com))  
To connect to an Oracle Real Application Cluster (RAC) database, use the format:  
DESCRIPTION=(ADDRESS=(PROTOCOL=protocol)(HOST=host1)(PORT=port)) (ADDRESS=(PROTOCOL=protocol)(HOST=host2)(PORT=port)) (CONNECT_DATA=(SERVICE_NAME=service name))  
For example:  
(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=myhost1.company.com)(PORT=1521)) (ADDRESS=(PROTOCOL=tcp)(HOST=myhost2.company.com)(PORT=1521)) (CONNECT_DATA=(SERVICE_NAME=sales.company.com))) |

| Connect as | Database user name. |
| **Password** | User’s password to access the database. |

6. Click **Test** to verify the connection.
7. Click **OK**.

Business modelers see the new connection in Data Modeler right away and can start to model the data. See [Creating a Data Model](#).

### Securing Database Connections with SSL

Use SSL to secure communication between Oracle BI Cloud Service and any database you connect to. You must obtain and upload a wallet that contains SSL certificates, to enable SSL on your Oracle Database Cloud Service connections.

1. Open the Console.
2. Click **Connections**.
3. If you've not done so already, upload a wallet file containing SSL certificates to Oracle BI Cloud Service:
   a. Click **Upload Wallet**.  
      To update an existing wallet file, click **Replace Wallet**.
   b. Click **Browse** and locate the wallet file.  
      Select a valid **cwallet.sso** file.
   c. Click **OK**.
4. Enable SSL security on a database connection:
   a. Create or edit a database connection.
      See Managing Database Connections.
   b. In the Connection dialog, select Enable SSL.
   c. Click OK.

Deleting Unused Data Sources

Your service comes with a fixed storage quota for data files. From time to time, administrators might need to delete data sources on behalf of other users to free up storage space and enable the service to function properly. For example, a user uploads data files and then their account is disabled when they leave the company.

1. From the Home page, click Data Sources.
2. Click Data Source Storage at the bottom of the page.
   The data storage quota and the total amount of space used reflects the quota for the entire service.

You can see who has uploaded data files and how much storage they’re using.

3. To free up some space, click the Options menu for a user with files you want to delete.
4. Select one of the following options:
   - **Delete Private** to delete non-shared (private) data files.
   - **Delete All** to delete all data files.

### Uploading Data Models from Oracle BI Enterprise Edition 12c

Administrators can upload data models built with Oracle BI Enterprise Edition to Oracle BI Cloud Service. After uploading a data model file (.rpd) to the cloud, content authors can then build data visualizations, dashboards and analyses in the usual way.

**Tutorial**

**Topics**
- About Uploading Oracle BI Enterprise Edition Data Models to the Cloud
- Getting Your Data Model File Ready
- Uploading Data Models from a File (.rpd) Using Console
- Editing Data Models Uploaded to the Cloud

**Note:**

Administrators can use snapshots to migrate content, as well as data models from Oracle BI Enterprise Edition. See [Migrating Snapshot Data](#).

### About Uploading Oracle BI Enterprise Edition Data Models to the Cloud

If you've modeled your business data with Oracle BI Enterprise Edition, then you don't need to start from scratch in Oracle BI Cloud Service. Simply upload your data model file (.rpd) to Oracle BI Cloud Service and start exploring your data through visualizations, analyses, and dashboards.

Oracle BI Cloud Service lets you upload a data model file (.rpd) with:

- One or more data models
Connections to one or more instances of Oracle Database Cloud Service
You’ll need to validate your data model file and configure database connection details in Oracle BI Enterprise Edition.

Note:
Copy reports and dashboards built in Oracle BI Enterprise Edition to Oracle BI Cloud Service too.
See Uploading Content from a Catalog Archive in Using Oracle Business Intelligence Cloud Service

Getting Your Data Model File Ready for the Cloud
Take some time to ready your data model file (.rpd) for the cloud:
• **Verify that you’re using Oracle BI Enterprise Edition 11.1.1.6 or later**
• **Validate the data model file**
  Run consistency checks using Consistency Check Manager in Oracle BI Administration Tool.
• **Verify that the data model file includes connection details to an Oracle Cloud database instance**
  Review connection pool settings from Oracle BI Administration Tool:
  – **Data source name** must point to the Oracle Database Cloud Service where the data is stored.
  – **Call interface** must be Oracle Call Interface (OCI).
    If your data model file connects to multiple Oracle Database Cloud Service instances, check connection pool settings one-by-one.

When the data model file (.rpd) is ready, you can upload it to Oracle BI Cloud Service. Before doing so, back up your current data model in case you need to restore this version. See Uploading Data Models from a File (.rpd) Using Console.

During the upload, existing data model information in Oracle BI Cloud Service is deleted and replaced with content from the uploaded file. Data models uploaded from the file become available to content authors through the Subject Areas pane.
Editing Data Models Uploaded to the Cloud

You can’t edit data models created with Oracle BI Enterprise Edition through Oracle BI Cloud Service. Data Modeler is disabled when you upload data models to Oracle BI Cloud Service from a .rpd file.

If you want to improve or update the model, make your updates in Oracle BI Administration Tool, re-run consistency checks, and upload the data model file again. Each time that you upload a data model, you delete and replace the existing model with the newly uploaded content.

**Tip:**

Model changes, such as deleting columns from the model, can affect existing content. Take time to check existing analyses and dashboards after uploading model updates.

Getting Your Data Model File Ready

Take some time to ready your data model file (.rpd) for the cloud.

1. Verify that you’re using Oracle BI Enterprise Edition 11.1.1.6 or later.
2. Validate the data model file (.rpd).
   Run consistency checks using Consistency Check Manager in Oracle BI Administration Tool.
3. Verify that the data model file includes connection details to Oracle Database Cloud Service.
   Review connection pool settings from Oracle BI Administration Tool:
   - **Data source name** must point to the Oracle Database Cloud Service where the data is stored.
   - **Call interface** must be Oracle Call Interface (OCI).
   If your data model file connects to multiple Oracle Database Cloud Service instances, check connection pool settings one-by-one.
4. Disable subject areas that you don’t want to expose or any subject areas that don’t have a working connection.
   If connection information is missing, users see the message Fetch subject areas failed error when they view subject areas in Data Visualization.
5. Back up your cloud service, including the current data model, in case you need to restore this version.

When the data model is ready, you can upload it to Oracle BI Cloud Service.
Uploading Data Models from a File (.rpd) Using Console

Administrators can upload data models built with Oracle BI Enterprise Edition to Oracle BI Cloud Service. After migrating data models to the cloud, content authors can visualize data in the usual way.

When you upload data models from Oracle BI Enterprise Edition, you delete existing data model information in Oracle BI Cloud Service and replace it with content in the data model file (.rpd). The data models you upload become available to content authors through the Subject Areas pane.

1. Verify the data model file (.rpd) and database connections.
   See About Uploading Oracle BI Enterprise Edition Data Models to the Cloud

2. In Oracle BI Cloud Service, click Console.

3. Select Snapshots.

4. Take a snapshot of the current data model in case you need to restore this version.
   See Taking a Snapshot.

5. Click Replace Data Model.

6. Click Browse and select the data model file (.rpd) that you want to upload.

7. Enter the password for the file.

8. Click OK.

9. Go to the Home page, click Data Sources then Subject Areas to see the data models that you uploaded, available as subject areas in Oracle BI Cloud Service.

10. Optionally, if the data model file that you uploaded defines permissions and data filters, create matching application roles in Oracle BI Cloud Service for the data security to work in the cloud.

    a. Create application roles with exactly the same names as those defined in Oracle BI Administration Tool.

    b. Assign users (and user roles) to the application roles as required.

Editing Data Models Uploaded to the Cloud

Metadata developers make updates to local data model files (.rpd) using BI Administration Tool.

You can’t use Data Modeler in Oracle BI Cloud Service to edit data models originally created with Oracle BI Enterprise Edition. Data Modeler is disabled when you upload data models from a file.

If you want to improve or update the model, make your updates in Oracle BI Administration Tool, re-run consistency checks, and upload the data model file again. Each time that you upload a data model, you delete and replace the existing model with the newly uploaded content.
Managing Map Information

Administrators set up map information so users can visualize and interact with data through maps.

Topics
- Setting Up Maps for Dashboards and Analyses
- Editing Background Maps

Setting Up Maps for Dashboards and Analyses

As the administrator, you define how data columns modeled in Oracle BI Cloud Service display on maps. Once you have configured the map data, users can visualize data in analyses through map views.

Tutorial

Map views allow users to display data on maps in different formats and to interact with data. Oracle BI Cloud Service is configured with Oracle MapViewer, spatial boundary data, hosted maps, Oracle Database, and optionally Oracle Spatial. As an administrator, you must configure the metadata that defines the mapping between Oracle BI data and spatial data.

1. On the Home page, click Console.
2. Click OBI Classic Maps.
3. On the Layers tab, click Import Layers from the toolbar.
4. In the Import Layers dialog, select the connection in the Look in field and the layers that are needed for zooming and drilling. Click OK.
5. Back on the Layers tab, select a layer and click the Edit Layer button. In the Edit Layer dialog, associate layers with columns so that users can display data in the map view.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Specifies the name of the layer that is displayed to users who work with map views.</td>
</tr>
<tr>
<td>Location</td>
<td>Specifies which background map the layer originates from. Click <strong>Location</strong> to select a different layer.</td>
</tr>
<tr>
<td>Description</td>
<td>Specifies the description of the map layer. This description is displayed when users are editing map views and they hover over the layer name in the Map Formats area.</td>
</tr>
<tr>
<td>Layer Key</td>
<td>Specifies the column of spatial data that you can associate with data for Oracle BI Cloud Service. Each column value corresponds to a “shape” that originates from the background map. For example, a MY_CITIES layer might have a layer key called CITY. The default value is a &quot;best guess&quot;. Select the appropriate column from the list.</td>
</tr>
<tr>
<td>BI Key Delimiter</td>
<td>Available only when multiple columns are specified for one key. Specifies a single ASCII character (such as a comma or underscore) to function as a delimiter for combining the data columns in Oracle BI Cloud Service that form a key.</td>
</tr>
<tr>
<td>Geometry Type</td>
<td>Specifies whether the layer is a polygon, point, or line geometry layer. The type that you select affects the formatting that users can apply to the layer.</td>
</tr>
<tr>
<td>BI Key Columns Area</td>
<td>Specifies the columns of data in Oracle BI Cloud Service that you want to associate with the layer. You can have multiple columns associated with a single layer. You can select multiple columns from one subject area or from multiple subject areas. The columns and delimiter that you select must exactly match the name of the <strong>Layer Key</strong> value. Suppose the Layer Key value is STATE_CITY. You must select the STATE and CITY BI data columns and specify the underscore character in the <strong>BI Key Delimiter</strong> field.</td>
</tr>
</tbody>
</table>

Use the various options in this area:
- **Add** — Displays the list of available subject areas. Select a subject area and select all the data columns that you want to associate with the layer.
- **Delete** — Deletes the selected key column.
- **Edit** — Lets you edit the data columns associated with a layer.

When a content designer creates a map view, a default main map is selected as the basis for that map view. If at least one data column from the analysis is associated with a layer that is associated with a main map, then that main map is selected by default.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Show Qualified Names</td>
<td>Specifies whether to display the fully qualified name of the column in the BI Key Columns Area or simply the column name.</td>
</tr>
</tbody>
</table>

You use this dialog to associate layers with BI data. If you use a specific column in multiple subject areas, then you must associate it with a layer for each subject area.

**Note:**

Spatial features such as shape definitions are managed by database administrators for your service. If a shape geometry definition does not exist for a particular column value, then the shape cannot be shown on the map and might inhibit user interactions on the map.

6. Click **OK** to close the dialog.
7. Click the Background Maps tab, then click the **Import Background Maps** button.
8. In the Import Background Maps dialog, select the connection in the **Look in** field and the main maps to use, then click **OK**.

The connection that you select for the main map can be different from the connection for the layers or images.

9. See **Editing Background Maps** for the steps required to prepare the background maps.

After you’ve added background maps and map layers, you can use the information to create a static image for a map. The static image is displayed to content designers and users who work with map views.

See Editing Map Views in *Using Oracle Business Intelligence Cloud Service*.
Editing Background Maps

You edit background maps to ensure that users have a seamless experience with map views.

A background map is a non-interactive map that serves as a base for the map view. It might display a satellite image or a map with roads. The background map specifies the order of layers on the map view.

The ordering of map layers is very important. You must pay close attention to ensure that users have a seamless experience while navigating on the map (that is, drilling and zooming). In the Edit Background Map dialog, you assign each layer a minimum and maximum zoom range. Given that the map zoom slider can slide only from bottom to top vertically, the layers with lower minimum zoom levels are placed at the bottom of the slider. Ensure that the layer grid on the Interactive BI Layers section of the dialog follows a similar pattern, so that you place layers with lower minimum zoom levels at the bottom of the list.

Layer ordering becomes irrelevant when the zoom ranges of layers don't intersect on the scale. Ordering becomes very important when layers have a common minimum and maximum zoom range. Use care to ensure that detailed layers aren't hidden by the aggregated layers during drilling or zooming operations.

1. Click **Console**.
2. Click **Map Data**.
3. On the Background Maps tab, select a map, then click the **Edit Background Map** button to display the Edit Background Map dialog.
4. Specify the name and description of the map, which is displayed as a tooltip for the map when selecting a map from the list, when editing the map view.
5. The Location field displays the location of the background map in the data source. Click **Location** button to change to a different map. If you select a background map that includes a different number of zoom levels, then the zoom levels are automatically adjusted for the layers that are associated with the map by scaling their ranges.
6. Click the **Add Layers** button to display a list of the layers that have been imported on the Layers tab, then select the layers to add to the map. This button is unavailable when all layers from the Layers tab have been added to the background map.

When you add a layer that's part of the map definition, the layer displays at its default zoom levels. If the layer is not part of the map definition, then specify the zoom levels yourself.

The layers are listed from bottom to top, in terms of how they're applied to the map. A sample order is Countries, States, Cities. The lower level layers generally have the lower zoom levels. For example, if you have a States layer and a Cities layer, then include lower zoom levels for State than City.
7. Click the **Sort Layers By Zoom Level** button to list the layers in ascending or descending order based on visibility on the map. This button is unavailable when layers are listed in the proper order.

The sort order that's specified here does not affect the order in which layers are applied on the map. Instead, the sorting order affects the zoom levels. For example, the States layer might have zoom levels 1 through 3 and the Cities layer has zoom levels 4 through 9. The lower layers have the lower zoom level numbers. The zoom levels that you specify correspond to the tick marks on the zoom slider on the map.

You can include both layers that have been associated with a BI column by using the Edit Layer dialog and layers that have not been associated. Ensure that BI layers are ordered higher than non-BI layers. If a non-BI layer is ordered higher than any BI layers, then the non-BI layer is displayed on top of the lower BI layers on the map, which prevents the BI layers from being interactive.

8. Click the **Turn On Layer Visibility** or **Turn Off Layer Visibility** button to control the visibility of layers on the map. Use the buttons to indicate whether the layer is visible in the Preview map in this dialog only. The layer is still visible on a map view. You can modify the zoom levels for a layer with a visibility turned off.

9. Click a cell under a zoom level for a layer to affect the zoom level:

   - If you click a blue cell that's between other blue cells, then you see a popup menu with **Clear Before** and **Clear After** buttons, which allow you to change the zoom level in either direction. For example, if you click the cell for zoom level 4 and click the eraser on the right, then all cells to the right are cleared for that zoom level.

   - If you click a blue cell that at the end of a line of blue cells, then the cell turns white to indicate that it is no longer part of that zoom level.

   - If you click a white cell, then you increase the zoom level on either side of the existing blue cells. For example, suppose cells 4 through 6 are colored blue to reflect the zoom level. If you click in cell 2, then the zoom level becomes 2 through 6.

If you don't set any zoom levels for a layer, then that layer does not display on the map.
10. Click the action icon beside the layer name to display a menu from which you can make various selections:
   - **Delete** — Removes the layer from this background map. The layer continues to be available on the Layers tab and can be added to this area again.
   - **Move Up or Move Down** — Moves the layer up or down so you can specify the order in which layers are applied to the map.
   - **Reset to Default Visibility** — Resets the current visibility range for this layer as defined in the underlying map definition. If this layer is not natively associated with the map, then this option is disabled for that layer.

11. Use the yellow border that surrounds the column of boxes for a zoom level to determine which zoom level is currently displayed in the map area.

12. Use the panning and zooming controls to specify how the map is displayed to users. If you hover over the zoom slider, then you see tooltips that specify the names of the layers that are currently associated with that zoom level.

13. Click OK.

### Whitelisting Safe Domains

Whitelisting allows or approves access to specific content. For security reasons, you’re not allowed to add external content to reports or embed your reports in other applications unless your administrator considers it safe to do so. Only administrators can add safe domains to the whitelist.

1. Click **Console**.
2. Click **Safe Domains**.
3. To allow users to embed content from other domains in their BI reports, click **Add Domain** under **Allow importing from**.
4. To allow users to embed their BI reports in content located on other domains, click **Add Domain** under **Allow embedding in**.
5. Enter the name of the safe domain. Use formats such as:
   - www.example.com
   - *.example.com
   - https:
6. For any safe domain you allow content to be imported from, select the types of resources you want to allow and block any resource types you don’t consider safe.
7. To remove a domain, select it and click the **Delete** icon.

**Note:**

After adding a safe domain, you’ll need to sign out and sign back in if you want to access content from that source.

See Embedding External Images and Other External Resources in Your Content and Embedding Your Content in Other Applications in *Using Oracle Business Intelligence Cloud Service*. 
Managing How Content is Indexed and Searched

Administrators can set up how catalog content is indexed and crawled so that users find the latest content when they search. By default, the catalog is crawled once a day and all the shared folders are included. You can set up a different schedule to better suit your business and exclude any folders you don't want searched.

Topics
- Configuring Search Indexing
- Scheduling Regular Content Crawls
- Monitoring Search Crawl Jobs

Configuring Search Indexing

Content is crawled and indexed so people can quickly find content when they search.

1. Click Console.
2. Click Search Index.
3. To ensure users find the most recent information when they search for data model objects, in the Data Model pane, select Enable Data Model Crawl and set up a full crawl.
   
   See Scheduling Regular Content Crawls to change how often content is crawled.
4. To ensure users find the most recent information when they search for content saved in the catalog, in the Catalog pane, select Enable Catalog Crawl and set up a full crawl.
   
   See Scheduling Regular Content Crawls to change how often content is crawled.

To temporarily suspend indexing, deselect Enable Data Model Crawl and Enable Catalog Crawl.

Scheduling Regular Content Crawls

It's the administrator's job to select which folders to crawl and schedule when and how often to crawl the content.

1. Click Console.
2. Click Search Index.
3. Select Data Model or Catalog.
4. Schedule when to run the crawl. Click Select Date and Time and specify the month, year, time, and time zone.
5. Schedule how often to run the crawl. Enter values for Run Every and Frequency to choose the best interval between crawls.
   
   By default, a crawl runs once a day. The index updates automatically as users add or modify content.
6. For catalog crawls, select Index User Folders to include private user content in the index.
User folders are indexed by default. Deselect this option if you don’t want any content stored under user folders to be indexed or searched.

7. Select the folders you want to crawl and exclude any folders with content you don’t want others to find when they search.

First select **Index User Folders**, and then select either **Index** or **Don’t Index** from the Crawl Status list for each folder.

8. For **Languages**, select all the languages you want to create indexes for. Crawl results are added to the index in the languages that you specify. For example, if your company’s headquarters are in the United States, and you have offices in Italy, then you can choose English and italiano to create an indexes in both English and Italian.

9. Click the **Save** icon to save your changes.

**Monitoring Search Crawl Jobs**

Administrators can check the last time content was indexed and monitor the status of crawl jobs. You can stop any crawl job that is running or cancel the next scheduled crawl before it starts.

1. Click **Console**.

2. Click **Search Index**.

3. Click **Monitor Crawls**.

   The Crawl Job Status page shows information about the past, current, and the next scheduled crawl.

4. Look at the **Status** column to find out when the content was last crawled and when the next crawl is due.

5. Click **Cancel** to stop a crawl job that is Running or Scheduled.

**Monitoring Users and Activity Logs**

You can see information about any users who are currently signed to your service and troubleshoot report queries from the Manage Session page.

**Topics:**

- Monitoring Users Who Are Signed In
- Analyzing SQL Queries and Logs

**Monitoring Users Who Are Signed In**

You can see how many users are signed in to your service and view detailed information about each user from the Manage Session page.

1. Click **Console**.

2. Click **Sessions and Query Cache**.

   The Sessions section at the top of the page shows how many users are currently signed in (Total Number of Sessions).

   The table provides details about the users who are signed in:
### Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>User ID</td>
<td>The name that the user entered when they signed in.</td>
</tr>
<tr>
<td>Session ID</td>
<td>A unique identifier assigned by Oracle BI Cloud Service for each user session.</td>
</tr>
<tr>
<td>Browser Info</td>
<td>Information about the browser used to sign in.</td>
</tr>
<tr>
<td>Logged On</td>
<td>Time when the user signed in.</td>
</tr>
<tr>
<td>Last Access</td>
<td>Time stamp for the last activity for this user. This can be any kind of activity, such as switching from one page to another.</td>
</tr>
</tbody>
</table>

3. To monitor a particular user, select **Filter Cursors by Session**.

Information for this user displays in the Cursor Cache table. See *Analyzing SQL Queries and Logs*.

Click **Clear Filter** to show information for all users.

4. To change how messages are logged for a particular user, select a **Log Level** from the list.

By default, logging is disabled.

### Analyzing SQL Queries and Logs

Administrators can examine the underlying SQL query requests that are executed as people use the service.

1. Click **Console**.

2. Click **Sessions and Query Cache**.

The Cursor Cache section enables you to monitor and troubleshoot activity logs for users currently signed in to the service.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>A unique internal identifier that is assigned to each entry.</td>
</tr>
<tr>
<td>User</td>
<td>The name of the user who ran the analysis and last placed it into the cache.</td>
</tr>
<tr>
<td>Refs</td>
<td>The number of references to this entry since it was placed into the cache.</td>
</tr>
<tr>
<td>Status</td>
<td>The status of the analysis that is using this cache entry:</td>
</tr>
<tr>
<td></td>
<td>• Starting — The analysis is starting to run.</td>
</tr>
<tr>
<td></td>
<td>• Waiting on Parent — A view in the analysis is waiting for data to be returned for the query.</td>
</tr>
<tr>
<td></td>
<td>• Running — The analysis is currently running.</td>
</tr>
<tr>
<td></td>
<td>• Finished — The analysis has finished.</td>
</tr>
<tr>
<td></td>
<td>• Queued — The system is waiting for a thread to become available so the analysis can be processed.</td>
</tr>
<tr>
<td></td>
<td>• Canceling — The application is in the process of canceling the analysis.</td>
</tr>
<tr>
<td></td>
<td>• Error — An error occurred during the processing or running of the analysis. Look in the Statement column for information about the error.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Time</td>
<td>The time taken to process and run the analysis, displayed in one-second</td>
</tr>
<tr>
<td></td>
<td>increments. A value of 0s (zero seconds) indicates that the analysis took</td>
</tr>
<tr>
<td></td>
<td>under 1 second to complete.</td>
</tr>
<tr>
<td>Action</td>
<td>Links that you can click to affect the analysis:</td>
</tr>
<tr>
<td></td>
<td>• Cancel — Terminates the analysis. Is displayed for in-progress analyses.</td>
</tr>
<tr>
<td></td>
<td>The user running the analysis receives an informational message indicating</td>
</tr>
<tr>
<td></td>
<td>that the analysis was canceled by an administrator.</td>
</tr>
<tr>
<td></td>
<td>• Close — Clears the cache entry associated with this analysis. Is</td>
</tr>
<tr>
<td></td>
<td>displayed for completed analyses.</td>
</tr>
<tr>
<td></td>
<td>• View Log — Displays the log of a query executed for this analysis.</td>
</tr>
<tr>
<td></td>
<td>• Diagnostic — Displays an HTML page of diagnostic information that you</td>
</tr>
<tr>
<td></td>
<td>can share with Oracle Customer Support.</td>
</tr>
<tr>
<td>Last</td>
<td>The time stamp of the last time the cache entry for this analysis was used</td>
</tr>
<tr>
<td>Accessed</td>
<td>to satisfy an analysis.</td>
</tr>
<tr>
<td>Statement</td>
<td>The logical SQL statement that was issued for the analysis; or if the</td>
</tr>
<tr>
<td></td>
<td>analysis resulted in an error, information about the nature of the error.</td>
</tr>
<tr>
<td>Information</td>
<td>Usage tracking information (for example, what analysis contained the</td>
</tr>
<tr>
<td></td>
<td>query).</td>
</tr>
<tr>
<td>Records</td>
<td>The number of records in the result set that have been seen (for example,</td>
</tr>
<tr>
<td></td>
<td>50+ to indicate that 50 records have been seen but there are</td>
</tr>
<tr>
<td></td>
<td>additional records to be fetched or 75 to indicate that 75 records have</td>
</tr>
<tr>
<td></td>
<td>been seen and there are no more records to be fetched).</td>
</tr>
</tbody>
</table>

3. Optionally, click **Close All Cursors** to removes information in the Cursor Cache table.

4. Optionally, click **Cancel Running Requests** to cancel all requests that are running for analyses.

### Executing Test SQL Queries

Administrators can enter a SQL statement directly to underlying data sources. This feature is useful for testing and debugging. Results aren't formatted and you can't save SQL statements that you issue directly.

Not all SQL functions and procedures are supported, such as the NQSSetSessionVariables() procedure.

1. Click **Console**.
2. Click **Issue SQL**.
3. Enter the SQL statement. For example:

```sql
SELECT XSA('weblogic'."SalesTargets")."Columns"."E1 Sales Rep Name" s_1,
       XSA('weblogic'."SalesTargets")."Columns"."P4 Brand" s_2,
       XSA('weblogic'."SalesTargets")."Columns"."T03 Per Name Qtr" s_3,
       XSA('weblogic'."SalesTargets")."Columns"."Target Revenue" s_4
FROM XSA('weblogic'."SalesTargets")
ORDER BY 2 ASC NULLS LAST, 3 ASC NULLS LAST, 4 ASC NULLS LASTFETCH FIRST 65001 ROWS ONLY
```

4. Change the **Logging Level** if required.
Each user receives a default logging level. Select Default in this box to use your default level.

5. Specify whether to run the query against cached data.
   Deselect Use Oracle BI Presentation Services Cache to specify that you don’t want the query to use cached data. In general, avoid deselecting this box as disabling the cache has potential performance degradation issues.

6. Click Issue SQL.

Monitoring Metrics for Oracle BI Cloud Service

Administrators can view status and track usage metrics for Oracle BI Cloud Service from My Services.

From My Services you can see whether Oracle BI Cloud Service and its associated Oracle Database Cloud Service are available and working as expected (Up).

You can also track how many people are using Oracle BI Cloud Service by their role:

- **BI Consumers This Month** — Reports how many of the people who signed in to Oracle BI Cloud Service during the current calendar month can view and run analyses (have the BI Consumer application role).

- **BI Authors This Month** — Reports how many of the people who signed in to Oracle BI Cloud Service during the current calendar month can create analyses and dashboards (have the BI Content Author application role) or perform advanced content management tasks, such as exporting dashboards (have the BI Advance Content Author application role).

- **BI Author Modelers This Month** — Reports how many of the people who signed in to Oracle BI Cloud Service during the current calendar month can model data and load data using Data Loader (have the BI Data Model Author application role).

- **BI Admins This Month** — Reports how many of the people who signed in to Oracle BI Cloud Service during the current calendar month can administer Oracle BI Cloud Service and delegate privileges to others (have the BI Service Administrator application role).

See Viewing Service Details in My Services in Managing and Monitoring Oracle Cloud.

Restarting Your Service

If your service isn’t responding you can stop and restart your service.

**WARNING:**

Your service will be temporarily unavailable while your system restarts. Everyone using the service will be signed out and lose any unsaved work.

1. Open the Console.
2. Click Snapshots.
3. Click Manage, then Restart Service.
4. Click OK to confirm.
Wait for a moment while the system restarts.

5. If the restart is successful, click **OK**.

6. If the restart fails, click **OK**.
   
   a. If you defined one or more database connections for your service, ensure these databases are available.
   
   b. If you uploaded a data model or restored a snapshot containing a data model, make sure any initialization blocks inside the model don't take too long to execute as this can cause timeouts. Use the Admin Tool on the source system to open the data model and check the initialization blocks.
   
   c. If you're still having issues, contact Oracle Support Services.
Part V
Reference

This part provides reference information for Oracle BI Cloud Service.

Appendixes:

• Frequently Asked Questions
• Troubleshooting
• Expression Editor Reference
• Data Sync Reference
Frequently Asked Questions

This reference provides answers to common questions asked by administrators and business intelligence analysts responsible for loading and modeling data in Oracle BI Cloud Service.

Topics:

• Top FAQs for Data Loading
  – What data loading methods can I use?
  – How does Data Sync load data?
  – Can I load non-UTF8 data?
  – For the full list of data loading FAQs, see Top FAQs for Data Loading.

• Top FAQs for Data Modeling
  – Can I use the same data for different analyses?
  – After adding new columns to my source table, can I include the new columns in my data model?
  – In what situations should I create model objects based on source views?
  – Can I include columns from a different source table or view in my existing dimension table when it is based directly on a source table?
  – Can I include columns from a different source table in my existing dimension table when it is based on a source view?
  – Can I create a source view that is based on another source view?
  – Can I migrate my data model from one environment to another?

• Top FAQs for Managing Oracle BI Cloud Service
  – How do I upgrade my Oracle BI Cloud Service?
  – Can I track how many users sign in to Oracle BI Cloud Service?
  – Can I see how many users are currently signed in?
  – Where can I find the public key for my service?
  – Can I see the SQL generated by an analysis and analyze the log?
  – What happens to my content if I terminate my subscription to Oracle BI Cloud Service?
  – In my Database Service I see some tables called S_NQ_DSS_CREDENTIALS, S_NQ_DSS_FS_FILES, S_NQ_DSS_SERVICE_INFO. I'm not sure what they’re for. Is it OK to delete them?
Top FAQs for Data Loading

The top FAQs for loading data are identified in this topic.

**Can I use Data Sync to transform my relational data?**

Yes. You can use Data Sync to transform relational data if you’re loading data into either Database As A Service or an on-premises database that is configured using the ‘Oracle (Thin)’ connection type.

**How do I transform my relational data?**

Follow the workflow described in Typical Workflow for Transforming Data Using Data Sync.

**Can I use Data Sync to transform my Data Sets?**

No. This is not supported in Data Sync V2.3.

**What data loading methods can I use?**

- Use Data Sync to load data from flat files, relational tables, OTBI, Oracle Service Cloud, or JDBC data sources. See About Data Sync.
- Use SQL Developer to load data from relational tables (for example, an Oracle database). See Loading Relational Tables Using SQL Developer.
- Use programmatic methods, such as PL/SQL scripts or the REST API to automate data loading. See Automating Loading Data Using Programmatic Methods.

---

**Note:**

You load to Database Cloud Service using Data Sync with a conventional on-premises connection (that is, use the Oracle (Thin) or Oracle (OCI8) connection type in from Data Sync.

**How does Data Sync load data?**

Data Sync uses REST APIs to load data into the Database Schema Service integrated with Oracle BI Cloud Service. When Data Sync loads data directly to Database Cloud Service, it uses JDBC.

**Can I use the REST API to load data?**

Yes. The REST API is optimized for loading large volumes of data (thousands to millions of rows) from one or more sources. See Automating Loading Data Using Programmatic Methods.

**Can I drop tables I don’t want any more?**

Yes. Use Oracle Application Express SQL Workshop to drop tables in Database Cloud Service. You can also use Data Sync to drop tables.
Can I load non-UTF8 data?

Yes. Data Sync supports a range of data formats. For a full list, see https://docs.oracle.com/javase/7/docs/api/java/nio/charset/Charset.html.

Can I change the batch size Data Sync uses to load data?

- With Database Cloud Service, the answer is Yes. For direct load into Database Cloud Service, the batch size is 10,000, and for initial loads, Data Sync uses parallel writes, which means that at any given time about 20,000 records may be written to the target.

- With direct REST API calls, the answer is Yes. Up to a million rows per batch is supported. However, such large batch sizes are not recommended because they are more difficult to manage. We recommend a batch size of 3,000 records, and this is the default for Data Sync.

Can I change the number of parallel processes that Data Sync uses?

No. For Database Cloud Service and on-premises targets, this is set to ten.

Is there a limit to the number of rows that Data Sync can load?

There is no limit to the amount of data that Data Sync can load. The size of your data storage dictates how much data you can load. For data sets, the storage limit is 50MB.

Can I allow for errors?

Yes. You can specify how many bad or invalid records that Data Sync tolerates. Click View, then System Properties to set Max Number Of Errors.

Where do I control the amount of memory allocated to Data Sync?

The default setting for Data Sync is 2GB. Data Sync initiates a separate Java process every time it runs a job. The memory requirement for the main Data Sync process is controlled by the -xmx parameter in datasync.bat/.sh. For individual jobs, memory values are read from conf-shared/jvm_parameters.txt.

If you're loading large amounts of data, then you can allocate more memory. You can also reduce the memory allocation if the hardware has limited memory for the Data Sync process.

Can I have multiple instances of Data Sync running on the same machine?

Yes. Install Data Sync again in a different directory and run it from there.

Can I duplicate my Data Sync environment?

Yes. Install Data Sync again in a different directory. When you run Data Sync for the first time, you'll see an option to copy an existing environment.

Can I integrate Data Sync jobs with other enterprise processes?

Yes. You can configure Data Sync to start jobs with external triggers – either in the form of a file or a database polling mechanism.
Can multiple users run Data Sync at the same time?
No. Data Sync is a single user tool. You can install Data Sync on a machine that allows for remote access by multiple users and start Data Sync whenever the machine starts up. However, you must avoid having multiple users access Data Sync at the same time. Remote users run `datasyncclient.bat/.sh` to access the environment.

Can I load data to multiple targets?
Yes. There’re two ways to do this.

- **Use a new job** — By default, Data Sync loads data to the default connection named TARGET. To upload data to a different connection:
  1. In the Connections view, create a new connection to the extra data target. For example, create a new connection named TARGET2 with connection details for your new target.
  2. In the Jobs view, navigate to Jobs, then Job.
  3. Click New, provide a job name (for example, RightNow-Job2), and click Next. The New Job dialog displays the default source and target connections for the project.
  4. In the Data Source column, select TARGET.
  5. Click the Override With column next to TARGET to display a list of available connections, and select the new target connection that you created in step 1 (for example, a connection named TARGET2).
  6. In the Project or Jobs workarea, click Run Job, and select the new job (for example, RightNow-Job2).

- **Use a second Data Sync installation on the same machine:**
  1. Install Data Sync again in a different directory.
  2. Export the metadata from the first environment (click Tools then Export) and import into the second (click Tools then Import).
     When you import metadata for the first time, select Logical and System. Subsequent imports need only Logical to be checked. In addition, select the Truncate option.
  3. In the second environment, configure the connection named TARGET to a different URL or database.
     By using a different URL or database target, you can isolate changes, which might be useful for testing.

Can I make the Data Sync process start automatically when a Windows machine is rebooted?
Yes. See Can I configure Data Sync as a Windows Service?.

Can I stop Data Sync creating or altering tables automatically?
Yes. Click View, System Properties, and set Automatically Create/Alter Tables to false.
Can I drop/alter/create just the schema objects from Data Sync?
Yes. Right-click on a table, and select Drop/Create/Alter Tables/DataSets.

Can I load data from multiple sources?
Yes. Create as many sources as you want and define a separate data flow for each source. You might find it more convenient to create a different project for each source. Alternatively, you can combine flows from different sources in a single project.

Can I load a subset of the tables in my project?
No. The granularity of a job is the project itself. All tables in a project are loaded when you run a job.

One workaround is to deactivate the tables that you do not want to load. You deactivate tables using the Inactive option on the Relational Data tab, File Targets tab (File Data), or Pluggable Source Data tab, depending on the type of the source. Another workaround is to create a separate project that only includes the subset of tables that you want to load.

I want to start my Data Sync install again from scratch? Can I do that?
Yes. If you have a test or proof of concept project that you want to throw away and start all over again, open a command window, navigate to the Data Sync installation directory, and run:

On Windows:
<code>datasync.bat -clean</code>

On UNIX:
<code>./datasync.sh -clean</code>

I have multiple jobs. Can I make one job run as soon as another job finishes?
Yes. At the beginning and end of a job, a file is created in the log\jobSignal directory. You can use these files to trigger another job. For information about configuring a job to listen for file signals, see Help: Triggering One Job After Another Automatically. You can use the same files for any other process to listen for when the job starts and ends.

Can I configure Data Sync as a Windows Service?
Yes. You can configure the Data Sync server on a Windows machine to keep running after you log off and to start automatically when the machine reboots. Use the Windows Task Scheduler (taskschd.msc) to create a new task and set these options:

- On the General tab:
  - Enter the user account that will run the task (this will typically be a network account).
  - Select Run whether user is logged on or not.
  - Select Run with highest privileges.

- On the Triggers tab:
  - Create a new trigger.
In the **Begin the task** list, select **At Startup**.

- On the **Actions** tab:
  - Create a new action.
  - In the **Program/Script** field, enter the full path and name of the bat file for the Data Sync server – that is, `<Data Sync install directory>\startserver.bat`.
  - In the **Start in (optional)** field enter `<Data Sync install directory>`.
- On the **Conditions** tab, review the default values and set field values as appropriate.
- On the **Settings** tab, review the default values and set field values as appropriate, then click **OK**.
- When prompted, enter the password of the user account.
  
  If you see the error “A specified logon session does not exist”, you will have to use a network account. For more information, see [https://blogs.technet.microsoft.com/askperf/2012/04/18/task-scheduler-error-a-specified-logon-session-does-not-exist/](https://blogs.technet.microsoft.com/askperf/2012/04/18/task-scheduler-error-a-specified-logon-session-does-not-exist/).

When your machine starts up, this task launches the Data Sync server. You won’t see the Data Sync sever as an icon in the system tray. Instead you’ll see a javaw.exe process in the Windows Task Manager (Processes tab).

To start the Data Sync client, double-click **datasyncClient.bat**.

To shut down the Data Sync Server, double-click **stopserver.bat**.

**Can I deploy Data Sync on a Cloud Compute Node?**

Yes. You need a VNC session to see the Data Sync tool. This option enables a cloud-to-cloud data replication solution for SaaS sources supported by Data Sync, without needing an on-premises server to host Data Sync.

**Can I monitor incomplete data loading jobs in all of my projects**

Yes. Click the server status icon in the top right-hand corner of the screen to display the Cross-project Current Jobs dialog.

---

**Top FAQs for Data Modeling**

The top FAQs for data modeling are identified in this topic.

**Can I use the same data for different analyses?**

Yes. You can create source views that expose the same source columns in different contexts. You use views to include the same source objects in multiple dimensions. For example, to use time data for both the Order Date and Ship Date dimensions, create two views based on the time source table, `time_order_date_v` and `time_ship_date_v`. The views can then be used as sources for the Order Date and Ship Date dimensions.

**After adding new columns to my source table, can I include the new columns in my data model?**

Yes. You can include newly added source columns. To include the new columns, select **Synchronize with Database** from the **Actions** menu for the appropriate fact.
In what situations should I create model objects based on source views?

Always create a source view when you think that you might want to perform subsequent changes, such as extending model objects, creating filters, and adding calculations. Creating a model based on source views provides greater flexibility than using source tables directly.

Can I include columns from a different source table or view in my existing dimension table when it is based directly on a source table?

Yes. It's easy to add columns from another source table or view to an existing dimension table. To do this, drag and drop the table or view on to your existing dimension table. See Adding Columns from Another Source to a Dimension Table.

Can I include columns from a different source table in my existing dimension table when it is based on a source view?

Yes. There are two ways you can do this. You can drag and drop the table on to the dimension table to include the columns. Alternatively, edit the view to include the new source columns, and synchronize your dimension table with your changes to the database. Synchronization identifies new columns in the view and adds them to your dimension table.

Can I create a source view that is based on another source view?

Yes. To do this, drag and drop the source view to the Columns area of the Overview tab when creating the view, or select a source view from the Add Columns dialog as your source.

For example, assume that you have both time and time_fiscal source tables. You created a view called time_v that combines time and time_fiscal. You want to create multiple dimensions that are based on time data, such as Order Day and Ship Day. You first create the Order Day dimension based on time_v, and then you create a separate view on top of time_v to create Ship Day. (Note that another option is to create a parallel view called ship_day_v that also combines time and time_fiscal.)

Can I migrate my data model from one environment to another?

Yes. To do this, take a snapshot of your data model and migrate it to the new environment. See Taking Snapshots and Restoring.

Top FAQs for Managing Oracle BI Cloud Service

The top FAQs for managing Oracle BI Cloud Service are identified in this topic.

How do I upgrade my Oracle BI Cloud Service?

Everyone who subscribes to Oracle BI Cloud Service receives automatic upgrades as soon as new versions become available. You don't need to request an upgrade or take any actions yourself. To find out about recent new features and enhancements, see What's New?
Can I track how many users sign in to Oracle BI Cloud Service?

Yes. Administrators can monitor how many users sign in to the service. Click Business Intelligence or Database from the Platform Services tab in the Oracle Cloud My Services page. See Viewing Service Details in Managing and Monitoring Oracle Cloud.

Can I see how many users are currently signed in?

Yes. Display the Home page, click Console, and then click Sessions and Query Cache. See Monitoring Users Who Are Signed In.

Where can I find the public key for my service?

Display the Home page, click Console, Connections, and then click Get Public Key.

Can I see the SQL generated by an analysis and analyze the log?

Yes. Display the Home page, click Console, and then click Sessions and Query Cache. See Analyzing SQL Queries and Logs.

What happens to my content if I terminate my subscription to Oracle BI Cloud Service?

When you terminate your subscription, Oracle takes a snapshot of the latest data model, catalog content, and application roles. The snapshot is sent to you in a password-protected Oracle Business Intelligence archive (.bar) file. If you subscribe to Oracle BI Cloud Service in the future, you can import content from this archive file. The password that you need to upload the archive file to another service is IdentityDomainName_ServiceName. See Uploading Snapshots and Restoring from a Snapshot.

In my Database Service I see some tables called S_NQ_DSS_CREDENTIALS, S_NQ_DSS_FS_FILES, S_NQ_DSS_SERVICE_INFO. I'm not sure what they’re for. Is it OK to delete them?

Data Visualization uses these tables. Do not modify, update, or drop these tables:

- S_NQ_DSS_CREDENTIALS
- S_NQ_DSS_FS_FILES
- S_NQ_DSS_SERVICE_INFO
Troubleshooting

This topic describes common problems that you might encounter preparing data in Oracle BI Cloud Service and explains how to solve them.

Topics:

- Troubleshooting General Issues
  - I can't sign in
  - I'm having trouble resetting my password
  - I can't access certain options from the Home page
  - I see a performance decrease when using Mozilla Firefox
  - I'm having trouble uploading data from a spreadsheet (XLSX) exported from Microsoft Access

- Troubleshooting Data Loading Issues
  - I can't start Data Sync
  - I can't connect Data Sync to my database
  - I get errors when loading data using Data Sync
  - Data Sync isn't reading my CSV file correctly
  - Data Sync isn't reading dates and times correctly
  - I can't connect Data Sync to my service

- Troubleshooting Data Modeling Issues
  - I can't see any tables or views in Data Modeler
  - I can't see the left pane in Data Modeler
  - I can't edit any objects in Data Modeler
  - I can't lock the data model
  - I can't publish the data model
  - Why must I use the SQL Query tab to edit a join or filter for a view?
  - I see the message: Cluster error-No active server node found

- Troubleshooting Administration Issues
  - I can't access options in the Console
  - I can't upload my snapshot
Troubleshooting General Issues

This topic describes common problems that you might encounter and explains how to solve them.

I can't sign in to Oracle BI Cloud Service

You're likely trying to sign in using the incorrect credentials. You must sign in to Oracle BI Cloud Service using the Oracle Cloud Identity Domain credentials that were mailed to you from Oracle or provided by your administrator. You can't sign in to Oracle BI Cloud Service using your account credentials for Oracle.com.

I'm having trouble resetting my password

When you sign up to use Oracle BI Cloud Service, you get an e-mail with a temporary password. Be careful if you copy and paste this password. If you accidentally include a blank space at the start or end of it when copying, then the password won't be recognized when you paste it in. Make sure that you paste only the password without any blank spaces.

I can't access certain options from the Home page

Check with your administrator to ensure that you have the correct permissions to access the options that you need.

I see a performance decrease when using Mozilla Firefox

If you use Mozilla Firefox and notice a decrease in the performance of the cloud service, then ensure that the Remember History option is enabled. When Firefox is set to not remember the history of visited pages, then web content caching is also disabled, which greatly affects the performance of the service. See Firefox documentation for details on setting this option.

I'm having trouble uploading data from a spreadsheet (XLSX) exported from Microsoft Access

Open your spreadsheet in Microsoft Excel and resave it as an Excel Workbook (*.xlsx).

When you export spreadsheets from other tools the file format can vary slightly. Saving your data again from Microsoft Excel can fix this.

Troubleshooting Data Loading Issues

This topic describes common problems that you might encounter when loading data and explains how to solve them.

Loading Data Using Data Sync

I can't start Data Sync

Verify that you installed Java JDK 1.7 or later and configured Data Sync's config.bat/.sh to point to this Java home. Data Sync doesn't work with JRE. Data Sync requires the JavaDB installed with JDK.
To check that you have JavaDB, look for a subdirectory named `db` in your Java install directory.

**I can't connect Data Sync to my database**

Data Sync ships with an Oracle 11.2 version of the JDBC driver. For all other supported databases, you must copy the appropriate JDBC driver to the `datasync\lib` directory.

Some vendors install multiple versions of the JDBC drivers compatible with different Java versions. Use only the JDBC driver files that relate to the Java version that Data Sync is configured to use. Oracle recommends that you do not have multiple versions of the JDBC drivers.

**I get errors when loading data using Data Sync**

Data Sync displays some status and error information on the Jobs tab and publishes more detailed information to log files:

- Current Jobs tab
  - Run Status
  - Status Description
- Run logs (.log)
  
  A log file for each job is saved in a separate directory under the `\log` directory. The naming convention used is `<Job Name>.<Process ID>`.
- Bad records (.bad)
  
  Errors caused by bad records are logged to a .bad file in the same directory as the log file. The .bad file logs the row that caused the issue and the error message for that row.

The Status Description field displays .log and .bad file details.

If you need more help, create a ZIP file of the contents of the log file directory and contact Oracle Support Services.

**I get error code 500 [BICS-DATALOAD] when loading data sets in Data Sync**

If you see this error when loading data sets, check that none of your data set files exceeds the maximum data set storage limit of 50MB.

**Data Sync isn't reading my CSV file correctly**

Find out which delimiter the CSV file is using and configure Data Sync to use the same delimiter. To verify the delimiter, in Project view display the File Data tab, display the Edit sub-tab, and click the **File** value to display the File Information dialog, and review the **Delimiter** option. If the delimiter is set correctly, then check that the **Codepage** setting matches the character set of your data file. The default character set in Data Sync is UTF-8, but your data file might be encoded with a different character set, for example ISO-8859-1 or US-ASCII.

The Java platform supports a range of character sets. See https://docs.oracle.com/javase/7/docs/api/java/nio/charset/Charset.html.
Data Sync isn't reading dates and times correctly

Data in CSV files is stored as strings. For Data Sync to recognize dates and times, specify the timestamp format that you want to use. Click Import Options and set the Timestamp option. If your format is not listed, then enter the format as it matches the data representation.

I can't connect Data Sync to my service

If your environment is using a proxy server, then you need to set some proxy options in Data Sync. Click View, System Properties, and then enter values for Proxy Host and Proxy Port.

Troubleshooting Data Modeling Issues

This topic describes common problems that you might encounter when modeling data and explains how to solve them.

I can't see any tables or views in Data Modeler

If you start Data Modeler and see no tables or views, then one of the following has occurred:

• There aren’t any tables in the database connected to your service. Use one of the supported data-loading tools to load some data.
• Data Modeler does not show the latest database objects. To see the latest objects, refresh the Database pane in Data Modeler.

I can't see the left pane in Data Modeler

The left pane in Data Modeler is collapsed. To display the left pane, click the Restore Pane icon displayed on the left of the page.

I can’t edit any objects in Data Modeler

You must always lock the data model before making any changes. Click Lock to Edit to lock the data model.

I can’t lock the data model

Check whether someone else locked the data model. If you’re an administrator, then you can override the lock. Otherwise, wait until the lock is released. For more information, see Overriding Locks in Locking the Data Model.

I can’t publish the data model

Check whether you have Data Modeler open in multiple browser tabs or multiple browser windows. If you do, close any additional browser tabs and windows running Data Modeler, and try publishing the model again. If you still get a publishing error, then restart the browser.

Why must I use the SQL Query tab to edit a join or filter for a view?

The message "Please use SQL Query tab to edit Joins/Filters" is displayed when you click the Joins tab or the Filters tab in the view editor for one of the following reasons:
• **Oracle BI Cloud Service can't parse the SQL query for the database view**

If you use only the Overview, Joins, and Filters tabs to edit a database view, then Oracle BI Cloud Service constructs a simple SQL query for you. If you decide to edit the SQL manually through the SQL Query tab, then simple updates are reflected back in the Overview, Join, and Filters tabs so you can use these tabs to further edit the view later on. If, however, you have used the SQL Query tab to make more advanced code changes, then you can't use the Overview, Joins or Filters tabs to further edit the view because Oracle BI Cloud Service can't verify your updates. For example, if you include:

- SQL aggregation functions, `GROUP BY` clause, `HAVING` clause
- `ORDER BY` clause
- `OR` keyword in `WHERE` clause
- `UNION` clause

• **Oracle BI Cloud Service can't access the database view**

If the problem persists, report the issue to your administrator. Your administrator can investigate connection issues relating to your database service.

I see the message: Cluster error-No active server node found

The instance might be down or the database might be locked. If the problem persists, then report the issue to an administrator.

---

**Troubleshooting Administration Issues**

This topic describes common problems that you might encounter when performing administration tasks and explains how to solve them.

**I can't access options in the Console**

If you see an "unauthorized" message or don't see an option in the Console, you probably don't have the BI Service Administrator application role. You must have the BI Service Administrator application role to access most Console options: Users and Roles, Snapshots, Connections, Safe Domains, Sessions and Query Cache, Issue SQL, Map Data, Search Index.

Ask an administrator to verify your permissions. See Assigning Application Roles to Users.

**I can't upload my snapshot**

You can only upload snapshots taken from Oracle BI Cloud Service. Check where the .bar file you're trying to upload was originally downloaded from.
This topic describes the expression elements that you can use in the Expression Editor in Data Modeler.

Topics:
- Data Model Objects
- SQL Operators
- Conditional Expressions
- Functions
- Constants
- Types
- Variables

Data Model Objects

You can use data model objects in expressions, like time levels, dimension columns, and fact columns.

To reference a data model object, use the syntax:

"Fact/Dimension Table Name"."Column Name"

For example: "Order Metrics"."Booked Amount"-"Order Metrics"."Fulfilled Amount"

The Expression Elements section includes only items that are relevant for your task, so not all fact tables and dimension tables might be listed. Similarly, time hierarchies are included only if the Time fact table is joined to the current table.

SQL Operators

SQL operators are used to specify comparisons between expressions.

You can use various types of SQL operators.

<table>
<thead>
<tr>
<th>Operator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BETWEEN</td>
<td>Determines if a value is between two non-inclusive bounds. For example: &quot;COSTS&quot;.&quot;UNIT COST&quot; BETWEEN 100.0 AND 5000.0. BETWEEN can be preceded with NOT to negate the condition.</td>
</tr>
<tr>
<td>IN</td>
<td>Determines if a value is present in a set of values. For example: &quot;COSTS&quot;.&quot;UNIT COST&quot; IN(200, 600, 'A')</td>
</tr>
<tr>
<td>IS NULL</td>
<td>Determines if a value is null. For example: &quot;PRODUCTS&quot;.&quot;PROD_NAME&quot; IS NULL</td>
</tr>
</tbody>
</table>
### Conditional Expressions

You use conditional expressions to create expressions that convert values.

The conditional expressions described in this section are building blocks for creating expressions that convert a value from one form to another.

---

<table>
<thead>
<tr>
<th>Expression</th>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CASE (If)</td>
<td>CASE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>WHEN score-par &lt; 0 THEN 'Under Par'</td>
<td>Evaluates each WHEN condition and if satisfied, assigns the value in the corresponding THEN expression. If none of the WHEN conditions are satisfied, it assigns the default value specified in the ELSE expression. If no ELSE expression is specified, the system automatically adds an ELSE NULL.</td>
</tr>
<tr>
<td></td>
<td>WHEN score-par = 0 THEN 'Par'</td>
<td></td>
</tr>
<tr>
<td></td>
<td>WHEN score-par = 1 THEN 'Bogey'</td>
<td></td>
</tr>
<tr>
<td></td>
<td>WHEN score-par = 2 THEN 'Double Bogey'</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ELSE 'Triple Bogey or Worse'</td>
<td></td>
</tr>
<tr>
<td></td>
<td>END</td>
<td></td>
</tr>
</tbody>
</table>

| CASE (Switch) | CASE Score-par |
|              | WHEN -5 THEN 'Birdie on Par 6' |
|              | WHEN -4 THEN 'Must be Tiger' |
|              | WHEN -3 THEN 'Three under par' |
|              | WHEN -2 THEN 'Two under par' |
|              | WHEN -1 THEN 'Birdie' |
|              | WHEN 0 THEN 'Par' |
|              | WHEN 1 THEN 'Bogey' |
|              | WHEN 2 THEN 'Double Bogey' |
|              | ELSE 'Triple Bogey or Worse' |
|              | END |

---

Note:
- In CASE statements, AND has precedence over OR
- Strings must be in single quotes

---
Functions

There are various types of functions that you can use in expressions.

Topics:

• Aggregate Functions
• Analytics Functions
• Calendar Functions
• Conversion Functions
• Display Functions
• Evaluate Functions
• Mathematical Functions
• String Functions
• System Functions
• Time Series Functions

Aggregate Functions

Aggregate functions perform operations on multiple values to create summary results.

<table>
<thead>
<tr>
<th>Function</th>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Avg</td>
<td>Avg(Sales)</td>
<td>Calculates the average (mean) of a numeric set of values.</td>
</tr>
<tr>
<td>Bin</td>
<td>Bin(UnitPrice BY ProductName)</td>
<td>Selects any numeric attribute from a dimension, fact table, or measure containing data values and places them into a discrete number of bins. This function is treated like a new dimension attribute for purposes such as aggregation, filtering, and drilling.</td>
</tr>
<tr>
<td>Count</td>
<td>Count(Products)</td>
<td>Determines the number of items with a non-null value.</td>
</tr>
<tr>
<td>First</td>
<td>First(Sales)</td>
<td>Selects the first non-null returned value of the expression argument. The First function operates at the most detailed level specified in your explicitly defined dimension.</td>
</tr>
<tr>
<td>Last</td>
<td>Last(Sales)</td>
<td>Selects the last non-null returned value of the expression.</td>
</tr>
<tr>
<td>Max</td>
<td>Max(Revenue)</td>
<td>Calculates the maximum value (highest numeric value) of the rows satisfying the numeric expression argument.</td>
</tr>
<tr>
<td>Median</td>
<td>Median(Sales)</td>
<td>Calculates the median (middle) value of the rows satisfying the numeric expression argument. When there are an even number of rows, the median is the mean of the two middle rows. This function always returns a double.</td>
</tr>
<tr>
<td>Min</td>
<td>Min(Revenue)</td>
<td>Calculates the minimum value (lowest numeric value) of the rows satisfying the numeric expression argument.</td>
</tr>
<tr>
<td>StdDev</td>
<td>StdDev(Sales)</td>
<td>Returns the standard deviation for a set of values. The return type is always a double.</td>
</tr>
<tr>
<td></td>
<td>StdDev(DISTINCT Sales)</td>
<td></td>
</tr>
<tr>
<td>Function</td>
<td>Example</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>--------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>StdDev_Pop</td>
<td>StdDev_Pop(Sales)</td>
<td>Returns the standard deviation for a set of values using the computational formula for population variance and standard deviation.</td>
</tr>
<tr>
<td></td>
<td>StdDev_Pop(DISTINCT Sales)</td>
<td></td>
</tr>
<tr>
<td>Sum</td>
<td>Sum(Revenue)</td>
<td>Calculates the sum obtained by adding up all values satisfying the numeric expression argument.</td>
</tr>
</tbody>
</table>

### Analytics Functions

Analytics functions allow you to explore data using models such as trendline and cluster.

<table>
<thead>
<tr>
<th>Function</th>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trendline</td>
<td>TRENDLINE(revenue, (calendar_year, calendar_quarter, calendar_month) BY (product), 'LINEAR', 'VALUE')</td>
<td>Fits a linear or exponential model and returns the fitted values or model. The numeric_expr represents the Y value for the trend and the series (time columns) represent the X value.</td>
</tr>
<tr>
<td>Cluster</td>
<td>CLUSTER((product, company), (billed_quantity, revenue), 'clusterName', 'algorithm=k-means;numClusters=1;maxIter=2;useRandomSeed=FALSE;enablePartitioning=TRUE', 5, 10)</td>
<td>Collects a set of records into groups based on one or more input expressions using K-Means or Hierarchical Clustering.</td>
</tr>
<tr>
<td>Outlier</td>
<td>OUTLIER((product, company), (billed_quantity, revenue), 'isOutlier', 'algorithm=mvoutlier')</td>
<td>This function classifies a record as Outlier based on one or more input expressions using K-Means or Hierarchical Clustering or Multi-Variate Outlier detection Algorithms.</td>
</tr>
<tr>
<td>Regr</td>
<td>REGR(revenue, (discount_amount), (product_type, brand), 'fitted', '')</td>
<td>Fits a linear model and returns the fitted values or model. This function can be used to fit a linear curve on two measures.</td>
</tr>
<tr>
<td>Evaluate_Script</td>
<td>EVALUATE_SCRIPT('filerepo://obiee.Outliers.xml', 'isOutlier', 'algorithm=mvoutlier;id=%1;arg1=%2;arg2=%3;useRandomSeed=False;', customer_number, expected_revenue, customer_age)</td>
<td>Executes an R script as specified in the script_file_path, passing in one or more columns or literal expressions as input. The output of the function is determined by the output_column_name.</td>
</tr>
</tbody>
</table>

### Calendar Functions

Calendar functions manipulate data of the data types DATE and DATETIME based on a calendar year.

<table>
<thead>
<tr>
<th>Function</th>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Current_Date</td>
<td>Current_Date</td>
<td>Returns the current date.</td>
</tr>
<tr>
<td>Current_Time</td>
<td>Current_Time(3)</td>
<td>Returns the current time to the specified number of digits of precision, for example: HH:MM:SS.SSS. If no argument is specified, the function returns the default precision.</td>
</tr>
<tr>
<td>Function</td>
<td>Example</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>---------------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Current_TimeStamp</td>
<td>Current_TimeStamp(3)</td>
<td>Returns the current date/timestamp to the specified number of digits of precision.</td>
</tr>
<tr>
<td>DayName</td>
<td>DayName(Order_Date)</td>
<td>Returns the name of the day of the week for a specified date expression.</td>
</tr>
<tr>
<td>DayOfMonth</td>
<td>DayOfMonth(Order_Date)</td>
<td>Returns the number corresponding to the day of the month for a specified date expression.</td>
</tr>
<tr>
<td>DayOfWeek</td>
<td>DayOfWeek(Order_Date)</td>
<td>Returns a number between 1 and 7 corresponding to the day of the week for a specified date expression. For example, 1 always corresponds to Sunday, 2 corresponds to Monday, and so on through to Saturday which returns 7.</td>
</tr>
<tr>
<td>DayOfYear</td>
<td>DayOfYear(Order_Date)</td>
<td>Returns the number (between 1 and 366) corresponding to the day of the year for a specified date expression.</td>
</tr>
<tr>
<td>Day_Of_Quarter</td>
<td>Day_Of_Quarter(Order_Date)</td>
<td>Returns a number (between 1 and 92) corresponding to the day of the quarter for the specified date expression.</td>
</tr>
<tr>
<td>Hour</td>
<td>Hour(Order_Time)</td>
<td>Returns a number (between 0 and 23) corresponding to the hour for a specified time expression. For example, 0 corresponds to 12 a.m. and 23 corresponds to 11 p.m.</td>
</tr>
<tr>
<td>Minute</td>
<td>Minute(Order_Time)</td>
<td>Returns a number (between 0 and 59) corresponding to the minute for a specified time expression.</td>
</tr>
<tr>
<td>Month</td>
<td>Month(Order_Time)</td>
<td>Returns the number (between 1 and 12) corresponding to the month for a specified date expression.</td>
</tr>
<tr>
<td>MonthName</td>
<td>MonthName(Order_Time)</td>
<td>Returns the name of the month for a specified date expression.</td>
</tr>
<tr>
<td>Month_Of_Quarter</td>
<td>Month_Of_Quarter(Order_Date)</td>
<td>Returns the number (between 1 and 3) corresponding to the month in the quarter for a specified date expression.</td>
</tr>
<tr>
<td>Now</td>
<td>Now()</td>
<td>Returns the current timestamp. The Now function is equivalent to the Current_TimeStamp function.</td>
</tr>
<tr>
<td>Quarter_Of_Year</td>
<td>Quarter_Of_Year(Order_Date)</td>
<td>Returns the number (between 1 and 4) corresponding to the quarter of the year for a specified date expression.</td>
</tr>
<tr>
<td>Second</td>
<td>Second(Order_Time)</td>
<td>Returns the number (between 0 and 59) corresponding to the seconds for a specified time expression.</td>
</tr>
<tr>
<td>TimeStampAdd</td>
<td>TimeStampAdd(SQL_TSI_MONTH, 12,Time.&quot;Order Date&quot;)</td>
<td>Adds a specified number of intervals to a timestamp, and returns a single timestamp. Interval options are: SQL_TSI_SECOND, SQL_TSI_MINUTE, SQL_TSI_HOUR, SQL_TSI_DAY, SQL_TSI_WEEK, SQL_TSI_MONTH, SQL_TSI_QUARTER, SQL_TSI_YEAR</td>
</tr>
<tr>
<td>TimeStampDiff</td>
<td>TimeStampDiff(SQL_TSI_MONTH, Time.&quot;Order Date&quot;,CURRENT_DATE)</td>
<td>Returns the total number of specified intervals between two timestamps. Use the same intervals as TimeStampAdd.</td>
</tr>
<tr>
<td>Function</td>
<td>Example</td>
<td>Description</td>
</tr>
<tr>
<td>----------------</td>
<td>----------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Week_Of_Quarter</td>
<td>Week_Of_Quarter(Order_Date)</td>
<td>Returns a number (between 1 and 13) corresponding to the week of the quarter for the specified date expression.</td>
</tr>
<tr>
<td>Week_Of_Year</td>
<td>Week_Of_Year(Order_Date)</td>
<td>Returns a number (between 1 and 53) corresponding to the week of the year for the specified date expression.</td>
</tr>
<tr>
<td>Year</td>
<td>Year(Order_Date)</td>
<td>Returns the year for the specified date expression.</td>
</tr>
</tbody>
</table>

### Conversion Functions

Conversion functions convert a value from one form to another.

<table>
<thead>
<tr>
<th>Function</th>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cast</td>
<td>Cast(hiredate AS CHAR(40)) FROM employee</td>
<td>Changes the data type of an expression or a null literal to another data type. For example, you can cast a customer_name (a data type of Char or Varchar) or birthdate (a datetime literal). Use Cast to change to a Date data type. Don't use ToDate.</td>
</tr>
<tr>
<td>IfNull</td>
<td>IfNull(Sales, 0)</td>
<td>Tests if an expression evaluates to a null value, and if it does, assigns the specified value to the expression.</td>
</tr>
<tr>
<td>IndexCol</td>
<td>SELECT IndexCol(VALUEOF (NQ_SESSION.GEOGRAPHY_LEVEL), Country, State, City), Revenue FROM Sales</td>
<td>Uses external information to return the appropriate column for the signed-in user to see.</td>
</tr>
<tr>
<td>NullIf</td>
<td>SELECT e.last_name, NULLIF(e.job_id, j.job_id) &quot;Old Job ID&quot; FROM employees e, job_history j WHERE e.employee_id = j.employee_id ORDER BY last_name, &quot;Old Job ID&quot;;</td>
<td>Compares two expressions. If they're equal, then the function returns null. If they're not equal, then the function returns the first expression. You can't specify the literal NULL for the first expression.</td>
</tr>
<tr>
<td>To_DateTime</td>
<td>SELECT To_DateTime ('2009-03-0301:01:00', 'yyyy-mm-dd hh:mi:ss') FROM sales</td>
<td>Converts string literals of dateTime format to a DateTime data type.</td>
</tr>
</tbody>
</table>

### Display Functions

Display functions operate on the result set of a query.

<table>
<thead>
<tr>
<th>Function</th>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BottomN</td>
<td>BottomN(Sales, 10)</td>
<td>Returns the n lowest values of expression, ranked from lowest to highest.</td>
</tr>
<tr>
<td>Filter</td>
<td>Filter(Sales USING Product = 'widget')</td>
<td>Computes the expression using the given preaggregate filter.</td>
</tr>
<tr>
<td>Function</td>
<td>Example</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>----------------------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Mavg</td>
<td>Mavg(Sales, 10)</td>
<td>Calculates a moving average (mean) for the last ( n ) rows of data in the result set, inclusive of the current row.</td>
</tr>
<tr>
<td>Msum</td>
<td>SELECT Month, Revenue, Msum(Revenue, 3) as 3_MO_SUM FROM Sales</td>
<td>Calculates a moving sum for the last ( n ) rows of data, inclusive of the current row. The sum for the first row is equal to the numeric expression for the first row. The sum for the second row is calculated by taking the sum of the first two rows of data, and so on. When the ( n )th row is reached, the sum is calculated based on the last ( n ) rows of data.</td>
</tr>
<tr>
<td>NTile</td>
<td>Ntile(Sales, 100)</td>
<td>Determines the rank of a value in terms of a user-specified range. It returns integers to represent any range of ranks. The example shows a range from 1 to 100, with the lowest sale = 1 and the highest sale = 100.</td>
</tr>
<tr>
<td>Percentile</td>
<td>Percentile(Sales)</td>
<td>Calculates a percent rank for each value satisfying the numeric expression argument. The percentile rank ranges are from 0 (1st percentile) to 1 (100th percentile), inclusive.</td>
</tr>
<tr>
<td>Rank</td>
<td>Rank(Sales)</td>
<td>Calculates the rank for each value satisfying the numeric expression argument. The highest number is assigned a rank of 1, and each successive rank is assigned the next consecutive integer (2, 3, 4,...). If certain values are equal, they are assigned the same rank (for example, 1, 1, 1, 4, 5, 5, 7...).</td>
</tr>
<tr>
<td>Rcount</td>
<td>SELECT month, profit, Rcount(profit) FROM sales WHERE profit &gt; 200</td>
<td>Takes a set of records as input and counts the number of records encountered so far.</td>
</tr>
<tr>
<td>Rmax</td>
<td>SELECT month, profit, Rmax(profit) FROM sales</td>
<td>Takes a set of records as input and shows the maximum value based on records encountered so far. The specified data type must be one that can be ordered.</td>
</tr>
<tr>
<td>Rmin</td>
<td>SELECT month, profit, Rmin(profit) FROM sales</td>
<td>Takes a set of records as input and shows the minimum value based on records encountered so far. The specified data type must be one that can be ordered.</td>
</tr>
<tr>
<td>Rsum</td>
<td>SELECT month, revenue, Rsum(revenue) as RUNNING_SUM FROM sales</td>
<td>Calculates a running sum based on records encountered so far. The sum for the first row is equal to the numeric expression for the first row. The sum for the second row is calculated by taking the sum of the first two rows of data, and so on.</td>
</tr>
<tr>
<td>TopN</td>
<td>TopN(Sales, 10)</td>
<td>Returns the ( n ) highest values of expression, ranked from highest to lowest.</td>
</tr>
</tbody>
</table>

**Evaluate Functions**

Evaluate functions are database functions that can be used to pass through expressions to get advanced calculations.

Embedded database functions can require one or more columns. These columns are referenced by \( %1 \) ... \( %N \) within the function. The actual columns must be listed after the function.
<table>
<thead>
<tr>
<th>Function</th>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Evaluate</td>
<td>SELECT EVALUATE('instr(%1, %2)', address, 'Foster City') FROM employees</td>
<td>Passes the specified database function with optional referenced columns as parameters to the database for evaluation.</td>
</tr>
<tr>
<td>Evaluate_Aggr</td>
<td>EVALUATE_AGGREGR_SLOPE(%1, %2), sales.quantity, market.marketkey)</td>
<td>Passes the specified database function with optional referenced columns as parameters to the database for evaluation. This function is intended for aggregate functions with a GROUP BY clause.</td>
</tr>
</tbody>
</table>

**Mathematical Functions**

The mathematical functions described in this section perform mathematical operations.

<table>
<thead>
<tr>
<th>Function</th>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abs</td>
<td>Abs(Profit)</td>
<td>Calculates the absolute value of a numeric expression.</td>
</tr>
<tr>
<td>Acos</td>
<td>Acos(1)</td>
<td>Calculates the arc cosine of a numeric expression.</td>
</tr>
<tr>
<td>Asin</td>
<td>Asin(1)</td>
<td>Calculates the arc sine of a numeric expression.</td>
</tr>
<tr>
<td>Atan</td>
<td>Atan(1)</td>
<td>Calculates the arc tangent of a numeric expression.</td>
</tr>
<tr>
<td>Atan2</td>
<td>Atan2(1, 2)</td>
<td>Calculates the arc tangent of (y/x), where (y) is the first numeric expression and (x) is the second numeric expression.</td>
</tr>
<tr>
<td>Ceiling</td>
<td>Ceiling(Profit)</td>
<td>Rounds a non-integer numeric expression to the next highest integer. If the numeric expression evaluates to an integer, the CEILING function returns that integer.</td>
</tr>
<tr>
<td>Cos</td>
<td>Cos(1)</td>
<td>Calculates the cosine of a numeric expression.</td>
</tr>
<tr>
<td>Cot</td>
<td>Cot(1)</td>
<td>Calculates the cotangent of a numeric expression.</td>
</tr>
<tr>
<td>Degrees</td>
<td>Degrees(1)</td>
<td>Converts an expression from radians to degrees.</td>
</tr>
<tr>
<td>Exp</td>
<td>Exp(4)</td>
<td>Sends the value to the power specified. Calculates (e) raised to the n-th power, where (e) is the base of the natural logarithm.</td>
</tr>
<tr>
<td>ExtractBit</td>
<td>Int ExtractBit(1, 5)</td>
<td>Retrieves a bit at a particular position in an integer. It returns an integer of either 0 or 1 corresponding to the position of the bit.</td>
</tr>
<tr>
<td>Floor</td>
<td>Floor(Profit)</td>
<td>Rounds a non-integer numeric expression to the next lowest integer. If the numeric expression evaluates to an integer, the FLOOR function returns that integer.</td>
</tr>
<tr>
<td>Log</td>
<td>Log(1)</td>
<td>Calculates the natural logarithm of an expression.</td>
</tr>
<tr>
<td>Log10</td>
<td>Log10(1)</td>
<td>Calculates the base 10 logarithm of an expression.</td>
</tr>
<tr>
<td>Mod</td>
<td>Mod(10, 3)</td>
<td>Divides the first numeric expression by the second numeric expression and returns the remainder portion of the quotient.</td>
</tr>
<tr>
<td>Pi</td>
<td>Pi()</td>
<td>Returns the constant value of pi.</td>
</tr>
<tr>
<td>Power</td>
<td>Power(Profit, 2)</td>
<td>Takes the first numeric expression and raises it to the power specified in the second numeric expression.</td>
</tr>
<tr>
<td>Radians</td>
<td>Radians(30)</td>
<td>Converts an expression from degrees to radians.</td>
</tr>
<tr>
<td>Rand</td>
<td>Rand()</td>
<td>Returns a pseudo-random number between 0 and 1.</td>
</tr>
<tr>
<td>Function</td>
<td>Example</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>-----------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>RandFromSeed</td>
<td>Rand(2)</td>
<td>Returns a pseudo-random number based on a seed value. For a given seed value, the same set of random numbers are generated.</td>
</tr>
<tr>
<td>Round</td>
<td>Round(2.166000, 2)</td>
<td>Rounds a numeric expression to $n$ digits of precision.</td>
</tr>
<tr>
<td>Sign</td>
<td>Sign(Profit)</td>
<td>This function returns the following:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• 1 if the numeric expression evaluates to a positive number</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• -1 if the numeric expression evaluates to a negative number</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• 0 if the numeric expression evaluates to zero</td>
</tr>
<tr>
<td>Sin</td>
<td>Sin(1)</td>
<td>Calculates the sine of a numeric expression.</td>
</tr>
<tr>
<td>Sqrt</td>
<td>Sqrt(7)</td>
<td>Calculates the square root of the numeric expression argument. The numeric expression must evaluate to a nonnegative number.</td>
</tr>
<tr>
<td>Tan</td>
<td>Tan(1)</td>
<td>Calculates the tangent of a numeric expression.</td>
</tr>
<tr>
<td>Truncate</td>
<td>Truncate(45.12345, 2)</td>
<td>Truncates a decimal number to return a specified number of places from the decimal point.</td>
</tr>
</tbody>
</table>

### String Functions

String functions perform various character manipulations. They operate on character strings.

<table>
<thead>
<tr>
<th>Function</th>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ascii</td>
<td>Ascii('a')</td>
<td>Converts a single character string to its corresponding ASCII code, between 0 and 255. If the character expression evaluates to multiple characters, the ASCII code corresponding to the first character in the expression is returned.</td>
</tr>
<tr>
<td>Bit_Length</td>
<td>Bit_Length('abcdef')</td>
<td>Returns the length, in bits, of a specified string. Each Unicode character is 2 bytes in length (equal to 16 bits).</td>
</tr>
<tr>
<td>Char</td>
<td>Char(35)</td>
<td>Converts a numeric value between 0 and 255 to the character value corresponding to the ASCII code.</td>
</tr>
<tr>
<td>Char_Length</td>
<td>Char_Length(Customer_Name)</td>
<td>Returns the length, in number of characters, of a specified string. Leading and trailing blanks aren’t counted in the length of the string.</td>
</tr>
<tr>
<td>Concat</td>
<td>SELECT DISTINCT Concat ('abc', 'def') FROM employee</td>
<td>Concatenates two character strings.</td>
</tr>
<tr>
<td>Insert</td>
<td>SELECT Insert('123456', 2, 3, 'abcd') FROM table</td>
<td>Inserts a specified character string into a specified location in another character string.</td>
</tr>
<tr>
<td>Left</td>
<td>SELECT Left('123456', 3) FROM table</td>
<td>Returns a specified number of characters from the left of a string.</td>
</tr>
<tr>
<td>Length</td>
<td>Length(Customer_Name)</td>
<td>Returns the length, in number of characters, of a specified string. The length is returned excluding any trailing blank characters.</td>
</tr>
<tr>
<td>Function</td>
<td>Example</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>-----------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Locate</td>
<td>Locate('d' 'abcdef')</td>
<td>Returns the numeric position of a character string in another character string. If the character string isn't found in the string being searched, the function returns a value of 0.</td>
</tr>
<tr>
<td>LocateN</td>
<td>Locate('d' 'abcdef', 3)</td>
<td>Like Locate, returns the numeric position of a character string in another character string. LocateN includes an integer argument that enables you to specify a starting position to begin the search.</td>
</tr>
<tr>
<td>Lower</td>
<td>Lower(Customer_Name)</td>
<td>Converts a character string to lowercase.</td>
</tr>
<tr>
<td>Octet_Length</td>
<td>Octet_Length('abcdef')</td>
<td>Returns the number of bytes of a specified string.</td>
</tr>
<tr>
<td>Position</td>
<td>Position('d', 'abcdef')</td>
<td>Returns the numeric position of strExpr1 in a character expression. If strExpr1 isn't found, the function returns 0.</td>
</tr>
<tr>
<td>Repeat</td>
<td>Repeat('abc', 4)</td>
<td>Repeats a specified expression n times.</td>
</tr>
<tr>
<td>Replace</td>
<td>Replace('abcdef1234', '123', 'zz')</td>
<td>Replaces one or more characters from a specified character expression with one or more other characters.</td>
</tr>
<tr>
<td>Right</td>
<td>SELECT Right('123456', 3) FROM table</td>
<td>Returns a specified number of characters from the right of a string.</td>
</tr>
<tr>
<td>Space</td>
<td>Space(2)</td>
<td>Inserts blank spaces.</td>
</tr>
<tr>
<td>Substring</td>
<td>Substring('abcdef' FROM 2)</td>
<td>Creates a new string starting from a fixed number of characters into the original string.</td>
</tr>
<tr>
<td>SubstringN</td>
<td>Substring('abcdef' FROM 2 FOR 3)</td>
<td>Like Substring, creates a new string starting from a fixed number of characters into the original string. SubstringN includes an integer argument that enables you to specify the length of the new string, in number of characters.</td>
</tr>
<tr>
<td>TrimBoth</td>
<td>Trim(BOTH '<em>' FROM 'abcdef</em>')</td>
<td>Strips specified leading and trailing characters from a character string.</td>
</tr>
<tr>
<td>TrimLeading</td>
<td>Trim(LEADING '_' FROM 'abcdef')</td>
<td>Strips specified leading characters from a character string.</td>
</tr>
<tr>
<td>TrimTrailing</td>
<td>Trim(TAILING '<em>' FROM 'abcdef</em>')</td>
<td>Strips specified trailing characters from a character string.</td>
</tr>
<tr>
<td>Upper</td>
<td>Upper(Customer_Name)</td>
<td>Converts a character string to uppercase.</td>
</tr>
</tbody>
</table>

System Functions

The USER system function returns values relating to the session.

It returns the user name you signed in with.

Time Series Functions

Time series functions are aggregate functions that operate on time dimensions.

The time dimension members must be at or below the level of the function. Because of this, one or more columns that uniquely identify members at or below the given level must be projected in the query.
### Function

<table>
<thead>
<tr>
<th>Function</th>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ago</td>
<td>SELECT Year_ID, Ago(sales, year, 1)</td>
<td>Calculates the aggregated value of a measure from the current time to a specified time period in the past. For example, <code>AGO</code> can produce sales for every month of the current quarter and the corresponding quarter-ago sales.</td>
</tr>
<tr>
<td>Periodrolling</td>
<td>SELECT Month_ID, Periodrolling (monthly_sales, -1, 1)</td>
<td>Computes the aggregate of a measure over the period starting ( x ) units of time and ending ( y ) units of time from the current time. For example, <code>PERIODROLLING</code> can compute sales for a period that starts at a quarter before and ends at a quarter after the current quarter.</td>
</tr>
<tr>
<td>ToDate</td>
<td>SELECT Year_ID, Month_ID, ToDate (sales, year)</td>
<td>Aggregates a measure from the beginning of a specified time period to the currently displayed time. For example, this function can calculate Year to Date sales.</td>
</tr>
<tr>
<td>Forecast</td>
<td>FORECAST(numeric_expr, ([series]), output_column_name, options, [runtime_binded_options])</td>
<td>Creates a time-series model of the specified measure over the series using either Exponential Smoothing or ARMIA and outputs a forecast for a set of periods as specified by <code>numPeriods</code>.</td>
</tr>
</tbody>
</table>

### Constants

You can use constants in expressions.

Available constants include Date, Time, and Timestamp. See also Current Date, Current Time, and Current TimeStamp.

<table>
<thead>
<tr>
<th>Constant</th>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Date</td>
<td>DATE [2014-04-09]</td>
<td>Inserts a specific date.</td>
</tr>
<tr>
<td>Time</td>
<td>TIME [12:00:00]</td>
<td>Inserts a specific time.</td>
</tr>
<tr>
<td>TimeStamp</td>
<td>TIMESTAMP [2014-04-09 12:00:00]</td>
<td>Inserts a specific timestamp.</td>
</tr>
</tbody>
</table>

### Types

You can use data types, such as `CHAR`, `INT`, and `NUMERIC` in expressions.

For example, you use types when creating `CAST` expressions that change the data type of an expression or a null literal to another data type.

### Variables

Variables are used in expressions.

You can use a variable in an expression. See Defining Variables.
Data Sync Reference

This topic includes links to reference information about installing and using Data Sync to load data for analysis.

Topics

• Installing and Updating Data Sync
• Help: About Data Sync
• Help: System Properties Dialog
• Help: Email Configuration Dialog and Recipients Dialog
• Help: Connections View
• Help: Cross-project Current Jobs
• Help: Creating and Modifying Tables
• Help: Creating and Modifying Data Sets
• Help: Jobs View
• Help: Job Schedules Dialog
• Help: Triggering Jobs from Other Tools
• Help: Triggering One Job After Another Automatically
• Help: Parameters/Execution Parameters dialog
• Help: Clearing the Cache After Uploading Data
• Help: Current Jobs Dialog and History Dialog
• Help: Consolidating Data from Multiple Sources
• Help: Uploading Data to Multiple Cloud Targets
• Help: Export Dialog and Import Dialog
• Help: Welcome Dialog
• Help: Pluggable Data Sources Dialog
• Help: Target Tables and Data Sets Dialog
• Help: Project Summary Dialog
• Help: Pre/Post SQL Processing Dialog
• Help: Patch Alerts Dialog
• Help: New Job Dialog
• Help: Mark as Completed Dialog
• Help: Parameters/Execution Parameters dialog
Installing and Updating Data Sync

To install Data Sync, you must meet the requirements and prerequisites, then unzip and run the application. Once installed, Data Sync notifies you of any available updates.

Topics

• About Security Guidelines and Requirements
• About Prerequisites, Supported Databases, and JDBC Requirements
• Installing Data Sync
• Starting Data Sync for the First Time
• Understanding Software Alerts in Data Sync
• Updating Data Sync

About Required User Accounts and Security Guidelines

To load data using Data Sync, you need user accounts with appropriate privileges.

What User Accounts Are Required?

For each Data Sync user, provision a user account that has the following:

• Application Role privileges for data loading, as specified in Giving Users Permissions to Upload Data with Data Sync. We recommend that you provision the specified Application Roles only, and restrict other access.
• Read privileges for each of your data sources.

Use these Data Sync user accounts in your Data Sync connections.

About Sensitive Information Stored By Data Sync

Data Sync stores sensitive information, including connection information for your data sources. We recommend that you run Data Sync in a controlled environment where the operating system and file system privileges are tightly controlled.

About Prerequisites, Supported Databases, and JDBC Requirements

Before installing Data Sync, you must have Java 1.7 or later Java Development Kit (JDK). On an on-going basis, apply any critical Java updates.

Note:

Data Sync doesn't work with Java Runtime Environment (JRE); you must have JDK.

Database Support

Data Sync supports the following databases:
• Oracle
• Microsoft SQL Server
• DB2
• Teradata
• MySQL
• Oracle TimesTen
• Generic JDBC with prepackaged drivers for MongoDB, Salesforce, Redshift, Hive and PostgreSQL
• Other sources that support JDBC
• Oracle Transactional Business Intelligence:
  – Oracle Financials Cloud
  – Oracle HCM Cloud
  – Oracle Procurement Cloud
  – Oracle Project Management Cloud
  – Oracle Sales Cloud
  – Oracle Supply Chain Management Cloud
• Oracle Service Cloud (RightNow)

**JDBC Drivers**

Data Sync is a Java application and uses JDBC to extract data from databases. Data Sync is installed with Oracle JDBC Version 12.1.0.2.0. If you’re using a different database or version, then you must replace the installed Oracle JDBC Version with the JDBC version specific to your database. To replace the installed JDBC, you copy the JDBC drivers to the \lib directory after you install Data Sync. For example, if your Oracle version is different, then copy the JDBC driver from your local Oracle installation.

<table>
<thead>
<tr>
<th>Vendor</th>
<th>JDBC Driver Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle</td>
<td>ojdbc7.jar</td>
</tr>
<tr>
<td>MySQL</td>
<td>Mysql-connector-java*.jar</td>
</tr>
<tr>
<td>Microsoft SQL Server</td>
<td>sqljdbc.jar</td>
</tr>
<tr>
<td>DB2</td>
<td>db2java.zip</td>
</tr>
<tr>
<td>TimesTen</td>
<td>tjjdbc6.jar, orai18n.jar, timestenjmsxla.jar, jms.jar, javax.jms.jar</td>
</tr>
<tr>
<td>Teradata</td>
<td>terajdbc4.jar, log4j.jar, teradata.jar, tdgssjava.jar, tdgssconfig.jar</td>
</tr>
</tbody>
</table>
Installing Data Sync

Download and install Data Sync in a few simple steps.

**Note:**

Data Sync stores sensitive information, including connection information to your on-premises databases. We recommend that you only install Data Sync in protected environments where the operating system and file system privileges are tightly controlled.

Before you install Data Sync, do this:

- Install Java Development Kit (JDK) 1.7 or later and apply any critical Java updates on an on-going basis.
  
  Data Sync doesn’t work with Java Runtime Environment (JRE). You must install JDK.

- Request permission to upload data. See Giving Users Permissions to Upload Data with Data Sync.

Install Data Sync in your environment.

1. Download Data Sync from Oracle Technology Network:
   

2. Unzip BICSDataSync_Vx_x.Zip to a directory with no spaces in its name.

3. Set your JAVA_HOME:
   
   a. Open config.bat (Windows) or config.sh (Linux or Unix).
   
   b. Replace @JAVA_HOME with the directory where JDK is installed.

   For example:

   ```
   set JAVA_HOME=D:\Java (on Windows)
   set JAVA_HOME=usr/java (on Linux or UNIX)
   ```

   If your directory name contains spaces you'll need to add double quotes around it.

4. Copy any database-specific JDBC drivers that you need to Data Sync’s \lib directory.

   Data Sync installs Oracle JDBC driver 11.2.x. If you want to connect to a different database (for example, Microsoft SQL Server or DB2) or if you want to use a different Oracle driver from the default version, obtain and manually copy the required files to the \lib directory. See also JDBC Drivers.

   Now you're ready to start Data Sync. See Starting Data Sync the First Time.

Starting Data Sync for the First Time

The first time you start Data Sync, you'll be asked to give your Data Sync repository a name and provide a password.
1. Start up Data Sync. Run `datasync.bat` (on Windows) or `datasync.sh` (on Linux/UNIX) from the directory where you installed Data Sync.

   The Data Sync icon displays in your system icon tray to show that the server is up and running.

2. Right-click the Data Sync icon and select **Start UI**.

3. Enter a **Logical Name** for the repository.

   This name is used to distinguish the repository in multi-repository environments. For example, you could name your repository *Development Environment* or *Production Environment*. You’ll see this name displayed as a tooltip on the system tray Data Sync icon and on the title bar in Data Sync.

4. Enter a password.

   Provide a password to access the client and select whether you want Data Sync to remember the password.

   **Note:**

   We recommend that you only install Data Sync in protected environments because Data Sync stores transactional system passwords.

5. Enter a name that describes your first project.

   Data Sync opens.

6. Set some basic system properties. See **Setting Default Options for Data Sync**.

   If your organization uses a proxy server to route calls to external websites, configure **Proxy Host** and **Proxy Port**.

7. Connect Data Sync to your data target. See **Specifying Connection Details for Your Cloud Service**.

8. Connect Data Sync to your data source. See **Specifying Connection Details for a Data Source**.

9. Optional. Set up email. See **Setting Up Email Notifications**.

**Starting and Stopping Data Sync**

To start Data Sync and its server, run `datasync.bat` (Windows) or `datasync.sh` (Linux/UNIX) from the directory where you installed Data Sync. The Data Sync icon displays in your system icon tray to show that the server is up and running.

- Select **Start UI** to open the Data Sync client.
- Select **Exit** to stop the Data Sync server.

Alternatively, run these files:

- `datasyncClient.bat.sh` opens the Data Sync tool (when server is running).
• stopserver.bat/sh stops the Data Sync server.

Reconfiguring Data Sync from the Beginning

To reset Data Sync to its default state and redo the setup process, run datasync.bat (Windows) or datasync.sh (Linux/UNIX) in a command window with the -clean option.

Uninstalling Data Sync

To uninstall Data Sync, delete the install directory.

Understanding Software Alerts in Data Sync

Data Sync notifies you if there is a new patch or software package.

If there is a new patch available or a new software package with additional functionality, Data Sync notifies you using the New Patch Alerts icon next to the Server Monitor icon. Prior to each load, Data Sync performs a version check, sending its current version to be checked against the one on the cloud. Depending on whether the versions match, the following actions occur:

• If the versions match, the data load continues.
• If the minor version is changed on the cloud, indicating an available patch, an optional alert is created and an email sent prior to continuing with the load. The alert is sent only once.
• When the major version is changed, indicating a new software package, an alert is created and an email sent. The data load halts, while informing you that a new version of the patch is required prior to doing any further data loads.

The number of unread alerts is indicated on the Alerts icon. Click the icon to view a list of alerts. If the icon is black there are no alerts, and it turns green when there are unread optional alerts, and red when there are unread mandatory alerts.

Updating Data Sync

You update Data Sync by performing a full installation and then migrating your environment.

New software updates are downloadable as compressed files that contain all content for a new installation of the software. To update the software, perform a full installation of the software as a new environment in a separate home directory, then migrate the environment from your existing installation.

Setting Up a New Environment with an Existing Environment Configuration

When you start the new environment after installing a patch or new version, you're prompted for environment configuration. Select Copy existing environment configuration and then specify the existing Data Sync environment's home directory. Data Sync repository and configuration files are copied to the new environment. If the new installation requires metadata upgrade, perform any upgrade after the files are copied.
Help: About Data Sync

With Data Sync, it's easy to upload on-premises data to your cloud database. Data Sync loads data directly from relational sources (tables, views, SQL statements), files (CSV and XLSX), and other sources such as OTBI, Oracle RightNow, Greenplum, MongoDB, Salesforce, Amazon Redshift, Hive, PostgreSQL, and more.

Some key terms and concepts:

- **Connection** — Defines data sources and target databases.
- **Project** — Workspace that defines and helps to organize your data uploads. For example, you could upload human resources and finance data under a single project (called “My Data”) or create two projects (called “My HR Data” and “My Finance Data”). Such partitions may be helpful if there is more than one user working on each system.
- **Job** — Uploads all the data defined in a project to your target Cloud database.

Help: Connections View

You set up connections to specify where your target Cloud service is and where your data sources are.

**Source/Targets list**

This list shows existing connections that have been setup for data sources and data targets.

- Edit the **TARGET** record and specify the connection details for your target Cloud service as described in ‘Specifying Connection Details For A Target Database’.
- To load data from a database, create a new record and specify the connection details as described in ‘Specifying Connection Details For A Source Database’.

**Note:** If you're loading data only from data files, then you don't need a connection in Data Sync. Go straight to the Project view, click the File Data tab, and specify your data file details.

For information about specifying connection details for a specific data type:

- JDBC sources, see Specifying Connection Details for Generic JDBC Sources.
- Oracle Service Cloud (RightNow), see Specifying Connection Details for Oracle Service Cloud (RightNow) .
- OTBI, see Specifying Connection Details for OTBI Data.
- NetSuite, see Specifying Connection Details for NetSuite Data.

Connection Details For A Target Database

<table>
<thead>
<tr>
<th>Field or Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Do not change the default name TARGET.</td>
</tr>
</tbody>
</table>
### Connection Details For A Source Database

<table>
<thead>
<tr>
<th>Field or Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Name</strong></td>
<td>Specify a short descriptive and environment-agnostic name such as CRM or HR to identify the connection in Data Sync. Avoid using instance-specific names such as host names, as the same connection can be configured against different databases in different environments (for example, development and production).</td>
</tr>
<tr>
<td><strong>Connection Type</strong></td>
<td>Select the type that best matches your data source.</td>
</tr>
<tr>
<td><strong>Table Owner</strong></td>
<td>Schema owner name. This is the user who owns the objects on the source schema. Make sure that the user has sufficient administration privileges on the reporting area that you want to load.</td>
</tr>
<tr>
<td><strong>Password</strong></td>
<td>Password for the database user/table owner.</td>
</tr>
<tr>
<td><strong>Service Name, TNS Name, Connection String, or Database Name, or ODBC Data Source</strong></td>
<td>Enter the values appropriate for your database. For an Oracle TNS Name, enter the TNS name that is specified in the tnsnames.ora file in the network\admin.</td>
</tr>
<tr>
<td><strong>Host</strong></td>
<td>Machine name or IP address of the machine where the database resides.</td>
</tr>
<tr>
<td><strong>Port</strong></td>
<td>Port number where the database listens (for example, 1521 is the default for an Oracle database).</td>
</tr>
<tr>
<td><strong>URL (Optional)</strong></td>
<td>A JDBC URL for the data source connection. The value in this field must conform to the database specifications. Use this option to specify a unique URL for a particular data source. For example, this option can be useful if this physical data source uses Oracle RAC and other data sources use a different database type.</td>
</tr>
<tr>
<td>Field or Element</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------</td>
<td>-----------------------------------------------------------</td>
</tr>
<tr>
<td>Driver (Optional)</td>
<td>The driver as described in the JDBC documentation.</td>
</tr>
</tbody>
</table>

**Note:** The JDBC driver version must match the database version. A version mismatch can lead to spurious errors during the data load process. Even using an Oracle database, if the version of the JDBC driver does not match that of the database, then you must download the compatible version of the JDBC driver from Oracle's website and place it in the lib directory.

For Oracle, it is preferable to use Thin type of connection. In some cases, such as RAC configurations, you might have to use the OCI8 type connection. In this case, also make sure the client library installed on the local machine matches with the database's version.

**Using Advanced Properties**

Use the **Generate** option on the top toolbar to create configurable properties for the type of data source that you're using. For example, for Oracle Service Cloud (RightNow), the properties are ‘Number of records to read at a time’ and ‘Default length of string datatypes’.

**Using Refresh Dates**

Use this tab to review the date that your data was last refreshed.

**Help: Cross-project Current Jobs**

Monitor all incomplete data loading jobs. This dialog is displayed in Data Sync when you click the server status icon in the top right-hand corner of the screen.

Use this dialog to monitor incomplete jobs in all of your projects, for example, jobs that are running, queued, or failed. If you find an incomplete job that you want to investigate further, make a note of the project name in the **Project** column and in the Jobs view, select that project and click Current Jobs. Here you can diagnose issues, review audit information, drill into individual task details, and restart or abort the job.

**Help: Current Jobs Dialog and History Dialog**

The Current Jobs tab shows in-progress or failed data load jobs. The History tab shows completed data load jobs.
Field or element | Description
---|---
**Current Jobs <List of job runs>** | Each time a job starts, a new job run is created and displayed in this list with the name and a process ID to uniquely identify the job run. The job run is displayed in this list until it is completed, in which case it is removed from this list and transferred to the History list of job runs. If a job run fails (with Run Status=Failed), it remains in the Current Jobs list until it is either:
  - Restarted and is successful (the Run Status is set to Completed).
  - Manually set to Completed. That is, you right click the job run and select **Mark as Completed**. If you do this, a new job run is created when you restart the job; otherwise the original job run is restarted.

**History <List of job runs>** | A list of all job runs for the current project.

**Edit** | Displays the status of the job run, which you can edit.

**Description** | Displays the details of the job run, including log files, messages, and various timestamps and metrics.

The run log files are stored in the log directory. One directory per run is created with a naming convention of CR_<Table/File Name>_<From Connection>_<To Connection>_<Timestamp>.log.

**Tasks** | The tasks and the details show important metrics, including start and end timestamps, number of rows processed, read and write throughput (number of rows processed per minute).

**Task Details** | Task Details tab elaborates on the line items for the data flow. The task details typically contain the following details:
  - Truncate Table: When the load strategy is set to Replace data in table or when loading a table (without the never delete data option) for the first time, truncate table is executed.
  - Drop Index: When a table is truncated, all the indexes on the table are dropped prior to the data load. The indexes whose 'Drop/Create Always' property is checked, always get dropped prior to the data load, even if the table is incrementally loaded.
  - Insert/Update/Append/Upsert: Depending on the load strategy, an appropriate command is used to load the data.
  - Create Index: When an index is dropped, it gets created after the data is loaded. Any new index that is registered which does not exist on the cloud also is created.
  - Analyze table: After data load and index creation, the table is analyzed.

**Audit Trail** | Displays the details of job run tasks.

**About Table Analysis**

To maximise performance, Data Sync only analyzes tables during initial loads and when tables are modified.

**Abandoning a Failed Run**

If for some reason you want to abandon a failed run, right-click its record in the Current Jobs tab and select **Mark as Completed**, which updates the job status to Completed. A subsequent request starts a new run.
Performing a Complete Reload of Data

To perform a complete refresh of your data (that is, reload all tables), on the Tools menu, select Re-extract and load all data. The next job run is performed in Full mode rather than Incremental mode, and reads and loads all data.

If you want to reload a single table, in the Connections view, select the table, display the Refresh Dates tab, click Re-Extract Data, and click Selected record only. In the Re-Extract Data dialog, click All Data and click OK. The next run will extract all data from the source table and reload the table.

Reloading Data that is Already Loaded

Data Sync remembers when a table is loaded into the database, which is typically the start time of the job. If your source database is in a different time zone, the value is adjusted accordingly. You can view this timestamp in the Refresh Dates tab of the Connections view.

For incremental loads, the run looks for records whose filter column value has changed since the timestamp of the last load. For example, take a load run after a load performed on June 1st, 2014 10:00 PM, only changes after that time would be extracted and loaded. You can modify the date to set it to a previous timestamp by clicking the Re-Extract Data button and providing a new value in the Re-Extract Data dialog box. For example, if you were to set the timestamp to March 1st, 2014 10:00 PM, the next run would look for data that changed since March 1st rather than June 1st.

Help: Email Configuration Dialog and Recipients Dialog

You can send data load reports by email to one or more recipients. Emails are sent from the email account that you configured on the Email Configuration dialog.

Email Configuration Dialog

<table>
<thead>
<tr>
<th>Field or element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>User Name</td>
<td>The user name for the email account.</td>
</tr>
<tr>
<td>Password</td>
<td>The password for the user name provided. (Only required if you select Needs Authentication.)</td>
</tr>
<tr>
<td>Email Server</td>
<td>The host name of the email server.</td>
</tr>
<tr>
<td>Email Server Port</td>
<td>The port where the email server listens.</td>
</tr>
<tr>
<td>Email Address</td>
<td>The email address of the user.</td>
</tr>
<tr>
<td>Needs Authentication</td>
<td>Specifies whether the corporate email server requires authentication.</td>
</tr>
<tr>
<td>Needs SSL</td>
<td>Specifies whether an SSL connection is required.</td>
</tr>
<tr>
<td>Send Test Email</td>
<td>Click to test the details by sending an email to the specified email account.</td>
</tr>
</tbody>
</table>
Recipients Dialog

<table>
<thead>
<tr>
<th>Field or element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the user to be notified.</td>
</tr>
<tr>
<td>Email Address</td>
<td>The email address where the notification is sent.</td>
</tr>
<tr>
<td>Inactive</td>
<td>Indicates whether the email notification for this recipient is active or inactive. When checked, the person is not included in the email broadcast.</td>
</tr>
</tbody>
</table>

Help: File Data Dialog

You can import data files from a root directory you define.

File Data dialog

<table>
<thead>
<tr>
<th>Field or element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;List of data loading projects&gt;</td>
<td>A list of existing data loading projects. To view details, click the file name in the File column or use the File field on the Edit tab to display file setup information.</td>
</tr>
<tr>
<td>Edit tab</td>
<td>This tab shows the project name and file name being loaded. Click the File field to display file setup information.</td>
</tr>
<tr>
<td>File Attributes tab</td>
<td>This tab shows the data columns identified in the data file. For example, if you're loading a spreadsheet, the attributes are the spreadsheet columns.</td>
</tr>
<tr>
<td>File Targets tab</td>
<td>This tab shows details of the target table, including table name and load strategy. To view or edit the load strategy, click the Load Strategy field. Click Column Mapping to view details of how source data columns map to target table columns, create lookups, and transform relational data.</td>
</tr>
</tbody>
</table>

New Source File wizard

<table>
<thead>
<tr>
<th>Field or element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Select File</td>
<td>Use this dialog to navigate to and select a local data file, for example, a CSV file or XLSX file. Click File Location and use the Open dialog to navigate to and select a data file. The File Name and Logical Name fields are populated automatically, but you can edit these if required. To clean up the data files after loading, click the Delete file(s) upon successful load. Before you use this option, for reference, make sure that you have copies of the data files stored in a different location.</td>
</tr>
</tbody>
</table>
Field or element | Description
--- | ---
**Import Options (for spreadsheet files)** | Use this dialog to specify information about your data file.
- **First line contains headers** - select this option if the first line of your spreadsheet contains headers that identify each column. If you clear this option, Data Sync will assign column names for you (COLUMN_1, COLUMN_2, and so on).
- **Timestamp format (Java style)** - specify the format of timestamp data in your data file.
- **Range of Cells** - (specific to spreadsheet files) specify the start cell and end cell of your data, including the header row if you have one. If the data file has a header row and you select the **First line contains headers** option but omit the header row from the range, then the first data row will be misinterpreted as the header.
- **Select sheets to be imported** - (specific to spreadsheet files) click to display the Choose Sheets dialog, which enables you to specify which sheets to load.
- **Number of lines to be sampled** - specify how many lines of data are analyzed by Data Sync to determine the syntax and structure.

**Import Options (for CSV files)** | Use this dialog to specify information about your data file.

---

**Configure Target** | Use this dialog to specify information about where to load your data.
- **Select an existing** - Click to display the Target Option dialog, which enables you to select an existing target table.
- **Create new** - Use this option to create a new target table with the name that you specify in the adjacent text box.

---

**File Information dialog**

Field or element | Description
--- | ---
**File Location** | Displays the location and file name of your data file. Click to select a different file.
**File Name** | (Editable) Displays the name of your data file.
**Logical Name** | Displays the target table name.

---

**Loading from Files**

In the system properties, you can define the root directory of data files for import by using the "Data File Root Directory" property. This specification allows the tool to open the directory as the default directory for choosing files, and ensures portability across different operating systems. It is recommended that all of the files are kept in this directory, as this property change helps in easily configuring the repository when moving the metadata to another environment without having to adjust the properties for each file entry.

Before importing files, it is important to review the file format by opening the file in a text editor to identify the delimiters, determine the number of lines to skip, and ensure that each record is on a new line.

**About File Format Requirements**

File format definitions and requirements include the following:
• You can specify a single character as delimiter. It can be a comma (",") or space (" ") or a tab ("\t" or "[tab]").

• When an attribute contains the delimiter as part of the value, the whole value needs to be double quoted.

• An attribute can contain new line characters, in which case it also needs to be double quoted.

• A file can have a banner, which can be ignored during load. However, the transition from header to data should predictably start from a certain line number.

• A file can have timestamp and date strings. However, only one format per file can be used. For example, if there is birth date and hire date, both need to be formatted in the same way. As an example, "yyyy-MM-dd" can be used to interpret timestamp "1968-01-01" as birth date, and "2010-06-31" as hire-date.

Refer to http://docs.oracle.com/javase/7/docs/api/java/text/SimpleDateFormat.html for various formatting possibilities.

About Error Handling and Logging

When a file is parsed and loaded, errors can result either while reading or writing. Read-related errors are mainly due to conversion of strings to an object of type integer, decimal, or timestamps. They can also result from bad formatting. For example, if an attribute contains the delimiter and it's not double quoted, or a line does not have as many attributes as the header. When there are bad records, the process fails.

Write-related errors can result from insufficient length or entering null into a not null attribute.

When such records are encountered, the errors are logged in a file in the log directory with the naming convention CR_<Table/File Name>_<From Connection>_<To Connection>_<timestamp>.bad.

This log file contains information including the line number of the bad record, the record itself, and what problems occurred when parsing the file.

Once the bad records are identified, fix the original file and rerun the process. If you are unsure about how to fix a record in the file at the location specified in the .bad log file, you can comment it out by adding "--" in front of the line in the file.

Importing Files

When you import files, Data Sync estimates the data types and other characteristics of the data and allows you to edit the values before creating a corresponding Oracle Business Intelligence Cloud Service data source in the Target Tables tab and registering the columns to load the data into. By default, up to 10,000 rows are taken as a sample to identify the data types. You can specify that a higher or lower number of rows be sampled, or choose -1 to read the whole file, though it may take longer to parse the whole file.

During import, the tool estimates the length of string attributes in the file so that the corresponding target table definition can accommodate the data. If it is a string representation, then the length is computed in increments of 50. If the length is more than 50% of the computed length, the next increment is used. The following table illustrates how the length is computed.
### Help: Export Dialog and Import Dialog

You can use the Import and Export features to migrate one Data Sync environment to another environment. For example, you might migrate Data Sync from a test environment to a production environment.

#### Before You Start

Data Sync behavior relating to the target folder is as follows:

- If the target folder is empty, Data Sync exports the metadata without a warning.
- If the target folder contains Data Sync metadata, the client issues a warning and you must click **OK** to proceed with the export. The export process replaces all content in the target folder.
- If the target folder has non-Data Sync metadata as well as Data Sync metadata, the client issues a warning, and you must click **OK** to proceed with the export. The export process replaces all content in the folder. All non-Data Sync metadata is deleted.
- If the target folder has only non-Data Sync metadata, Data Sync cannot export into the specified target folder.

#### Exporting Metadata

You use the export feature to create a copy of source system-specific metadata that you can use to migrate Data Sync. For example, you might export metadata from your test environment, and then import the metadata into your production environment.

1. In Data Sync, select **Export** from the **Tools** menu.
2. Select the directory to which you want to export metadata, or accept the default directory.
3. Select the appropriate applications that you want to export metadata for.
4. Select the appropriate categories of metadata you want to export:
   - Logical: Exports all information contained in the Project view.
   - System: Exports all information contained in the Connections view, except passwords for servers and database connections.
   - Run Time: Exports information about jobs and schedules contained in the Jobs view.
   - User Data: (Applicable to Data Sync standalone authentication only) Exports users, roles, and passwords.

---

#### Maximum length of string in sample data

<table>
<thead>
<tr>
<th>Maximum length of string in sample data</th>
<th>Computed length</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>50</td>
</tr>
<tr>
<td>27</td>
<td>100</td>
</tr>
<tr>
<td>55</td>
<td>100</td>
</tr>
</tbody>
</table>
Note:

When importing roles and passwords, if the encryption key in the target repository is different to the encryption key in the source repository, the roles and passwords will be unreadable.

5. Verify the export process by reviewing the log file `<Domain_Home>/log/export.log`. You can use the Overwrite Log File option to overwrite earlier export logs.

Importing Metadata

You can use the import feature to migrate source system-specific Data Sync metadata into a Data Sync environment. For example, if you previously exported metadata from your test environment, you might import the metadata into the production environment.

1. In Data Sync, select **Import** from **Tools** menu.

2. Select the directory from which you want to import metadata, or accept the default directory.

3. Select the appropriate applications for which you want to import metadata.

4. Select the appropriate categories of metadata you want to import:
   - **Logical**: Imports all information contained in the Project view.
   - **System**: Imports all information contained in the Connections view, except passwords for servers and database connections.
   - **Run Time**: Imports information about jobs and schedules contained in the Jobs view.

5. To import metadata into a blank repository or to completely replace selected categories of the current metadata in the repository, select **Truncate repository tables**. This option overwrites the content in the current repository. It also greatly increases the speed of the import process.

6. To import new records and update existing records, select **Update existing records**. If you don't select this check box, Data Sync inserts only new records. This option isn’t available if you select the **Truncate Repository Tables** option.

7. (Optional) Select **Enable bulk mode** to insert the imported metadata into the repository as an array insert. You should elect this option only if you also selected the **Truncate Repository Tables** option. This action increases the speed of the import process.

8. Click **OK**.

9. Verify the import process by reviewing the log file `<Domain_Home>/log/import.log`.

Help: Job Schedules Dialog

You schedule jobs to load your data regularly. For example, you might perform an incremental data load once per week. Before you start, you first create a job using the Jobs tab.

1. Select the **Job Schedules** tab to open the scheduler.

2. In the **Edit** tab, specify a short **Name** to identify the schedule in Data Sync.
3. Specify the following details:

<table>
<thead>
<tr>
<th>Field</th>
<th>What to specify</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Specify a short name to identify the schedule in Data Sync.</td>
</tr>
<tr>
<td>Job</td>
<td>Select the data load that you want to schedule by selecting a Job. Data Sync creates a default job when you create a project, named &lt;Project name&gt;-Job&lt;n&gt;. You can run this job, or run a different job that you created yourself.</td>
</tr>
<tr>
<td>Run Only Once</td>
<td>Select this option to load data once only.</td>
</tr>
<tr>
<td>Inactive</td>
<td>Select this option to deactivate the data load schedule.</td>
</tr>
<tr>
<td>Restart Failed</td>
<td>Select this option to reload data automatically if an error is detected. Data Sync will restart the failed job. If this option isn't selected, then you have to manually restart failed jobs or mark them as completed.</td>
</tr>
<tr>
<td>Run Once Only</td>
<td>Select this option to load the data once and hide recurrence options.</td>
</tr>
<tr>
<td>Recurrence Pattern</td>
<td>Specify how frequently you want to load your data. These options are only available when the Run Only Once option is not selected.</td>
</tr>
<tr>
<td>Start date/End date</td>
<td>Specify when to start and stop the data loads.</td>
</tr>
</tbody>
</table>

4. Click **Save**.

In the top pane, you can view and select schedules and view their status and other properties. In the Next Run column for a schedule, click the button to open a Date dialog box, in which you can specify a new date to run the scheduled job on. Upon this change, the schedule resumes under the recurrence pattern you specify when creating the schedule.

**Help: Jobs View**

A job is the unit of work you use to organize, schedule, execute, and monitor load processes. A run is an instance of a data loading job. For example, if you run a job twice, then you'll see two run records on the History tab.

You can use a job to load your data one time only (see Loading Data Using Data Sync) or regularly (see Refreshing Data Regularly).

Use:

- The Jobs sub-tab to create a data loading job that you can run once only or run regularly.
- The Job Schedules sub-tab to load data regularly using an existing job.
- The Current Jobs sub-tab to manage recently started data loads.
- The History sub-tab to review and manage completed data loads.
Starting a data load using a Job

To start a data load, display the Jobs tab, and use the lower tabs to change the default data load configuration settings. Click Run Job to start loading data.

If you’re loading data to DBaaS, then on the Edit tab use the Cloud Connection for Cache Purging option to specify the cache to purge after each job run. If you’re loading data to DBaaS and you leave the Cloud Connection for Cache Purging field blank, then your target data cache will not be purged, which means that it will take longer for the new data to appear in your BI reports.

More About Working with Jobs

When you create a project, Data Sync creates a default job for you named <Project Name>-Job1. In the Jobs view, you can view the default job or create new jobs. When you select a job, the Ordered Tasks sub tab lists the tasks which are run when the job is executed. A job is initially empty, with no ordered tasks. Upon the first run of a job the tasks are automatically computed and executed in order.

If more than one job is in an incomplete status (such as Running or Failed or Stopped), the job fails. If a job fails, you might have to manually marked the status as completed before you can restart it. To mark a run as completed, right-click the run entry and select Mark as Completed.

Restarting Jobs Automatically

Data loads to the cloud can fail due to network issues, and might succeed when the data load is rerun. If you want Data Sync to automatically retry upon failure, you can set the number of retries at the job level using the Jobs\Edit\# Retries field.

Refining Jobs

<table>
<thead>
<tr>
<th>What do I want to do?</th>
<th>Use this tab</th>
</tr>
</thead>
<tbody>
<tr>
<td>Review the order of tasks that are included in the data-loading job.</td>
<td>Ordered Tasks</td>
</tr>
<tr>
<td>Notify people automatically with a status email when data has been loaded.</td>
<td>Email Recipients</td>
</tr>
<tr>
<td>Specify data loading behavior that is specific to the type of data source being used.</td>
<td>Connectivity Parameters</td>
</tr>
<tr>
<td>Customize a data load or override a project-level parameter with a job-level parameter (known as an execution parameter).</td>
<td>Execution Parameters</td>
</tr>
</tbody>
</table>

Purging Run Details

To remove details of completed data loads from the History tab, select Tools, then Purge Run Details. Use the Purging Run History dialog to specify how much history data to remove. For example, select All completed runs to remove run records with a Run Status of Completed. To remove all information about each completed run, (for example, to minimize the Data Sync repository size), make sure that the Keep run definitions option is not selected. Don’t forget that deleted run information cannot be recovered. To keep a summary of each run but remove the tasks, task details, and audit trail details, then select the Keep run definitions option.
Help: Load Strategy Dialog

Before you load data, you define how the data will be loaded by choosing a load strategy.

About Load Strategies

A load strategy defines how your data is loaded from a specific data source into your target. When you choose a load strategy that incrementally loads the data, Data Sync requires you to define a user key to uniquely identify a record on the target side, and a DATE/TIMESTAMP based column which can be used to identify the incremental data. If an index is not available, then Data Sync prompts you to create an index.

An example load strategy

You have a table with CONTACT_ID as the unique identifier for any record, and a date column LAST_UPD whose value is updated to the current timestamp whenever a record is created or updated. Here, you would choose CONTACT_ID for user key and LAST_UPD column as the Filter.

When data is loaded for the first time, Data Sync issues a `SELECT * FROM CONTACT` statement. If the first load happened on January 1, 2014 at 10:00 AM, the subsequent load would issue the following SQL statement (Oracle syntax): `SELECT * FROM CONTACT WHERE LAST_UPD > TO_DATE('01-01-2014 10:00', 'MM-DD-YYYY HH24:MI')`. The record set then compares the CONTACT_ID value to the data already existing in the Oracle Business Intelligence Cloud Service schema. Any record without a match is inserted. Any records with a match are updated. It is important that the source system contains an index for the incremental filter column.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Replace data in table</td>
<td>Delete any existing data and reload data always. Also applies to loads where a unique key is not available. Does not require a primary key or filter column. Table is truncated before data load on each run. Any indexes are dropped prior to data load and recreated after load. Table is analyzed at the end to update statistics.</td>
</tr>
<tr>
<td>Append data to table</td>
<td>New data is added to the table without checking for any prior existence of data. Does not require a primary key or filter column. Table is never truncated. If registered indexes do not exist on the provisioned schema, they are created after the data load.</td>
</tr>
<tr>
<td>Update table (Add new records)</td>
<td>Requires a primary key or filter column. If the data with the user key is not available, then it is inserted, else the record is ignored. During initial run, the table is truncated before the first load. Indexes are dropped prior to data load, and recreated after load. The table is analyzed at the end to update statistics. During incremental runs, the table is not truncated. Data is applied incrementally, and any missing indexes are created. The table is analyzed at the end. If any index is marked as &quot;Always drop and create&quot;, those indexes do get dropped and created even during incremental runs.</td>
</tr>
<tr>
<td>Value</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Update table (Update existing records)</td>
<td>Requires a primary key or filter column. If the data with the user key is available, then it is updated, else it is ignored. During initial run, the table is truncated before the first load. Indexes are dropped prior to data load, and recreated after load. The table is analyzed at the end to update statistics. During incremental runs, the table is not truncated. Data is applied incrementally, and any missing indexes are created. The table is analyzed at the end. If any index is marked as “Always drop and create”, those indexes do get dropped and created even during incremental runs.</td>
</tr>
</tbody>
</table>

**Help: Mark as Completed Dialog**

Cancel the current data load by changing the status of the job run to completed.

To confirm that you do want to cancel the current data loading job run, enter the random code displayed into the text box, then click **Yes**. When you restart the job by clicking **Run Job**, Data Sync will create a new job run instead of restarting the failed job run. You can monitor the new job run on the **Current Jobs** tab.

**Help: New Job Dialog**

Specify a data source and data target for your new data loading job.

- **Job name**: Specify a unique name to identify the data loading job.
- **Data Source**: This column displays the connection name for your data source (for example, your RightNow data source) and the connection name for your data target. If you want to use the defaults specified, click **Finish**. If you want to change either the data source or data target for this job, use the **Override With** option.
- **Override With**: Use this option to change the data source or data target for this data loading job. For example, you want to perform a test run from a smaller RightNow data set with a connection named RightNow-Test01. Here, you click the **Override With** column for the RightNow table row, and select RightNow-Test01. When you run the new job, Data Sync will load from the smaller RightNow data set rather than the full data set.

Similarly, to test a data load, you might want to change the default data target to a different data target.

**Help: Parameters/Execution Parameters dialog**

You use parameters to customize your data loads at run time.

**Why should I use parameters?**

Parameters enable you to dynamically customize the way you load data. For example, if you want to load data from the previous one year, you might create a parameter named `NUM_YEARS_TO_EXTRACT` and set the value to 1. Then, you can use this variable in a query override (to reference a parameter, you prefix the parameter name with `%%`), for example:
SELECT * FROM MY_REVENUE WHERE CREATED < SYSDATE - (%NUM_YEARS_TO_EXTRACT *365)

If you want to change the number of years’ data to load, you don’t have to edit the SQL query, you simply use the Project > Parameters dialog to change the value.

You can also use parameters to configure your data loads with a number of commonly used runtime variables that provide job information, such as ETL_START_TIME, CURRENT_TIMESTAMP, and CURRENT_PROCESS_ID.

How do I define a parameter?

You can define parameters:

• at a Project level, using the Project\Parameter tab.
• at a Job level, using the Jobs\Job\Execution Parameters tab. Remember that an execution parameter overrides a project-level parameter with the same name.

You can override a project level parameter using an execution parameter with the same name set at the job level. For example, your project loads five years of data by default, but for a test environment you might want to load one year of data only. In this scenario you have a parameter named NUMBER_YEARS_TO_EXTRACT defined with the value 5 on the Project\Parameter tab. For the default job, TARGET is pointing to a production environment. You create a new job in the same project, and on the Jobs\Job\Execution Parameters tab, you create an execution parameter with the same name NUMBER_YEARS_TO_EXTRACT and set its value to 1.

Specify these details:

<table>
<thead>
<tr>
<th>Field or Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Specify a short name (less than 20 characters) with no spaces to identify the parameter in Data Sync.</td>
</tr>
<tr>
<td>Data Type</td>
<td>Select Text or Timestamp, depending on what runtime variable you want to attach to the parameter.</td>
</tr>
<tr>
<td>Load Type</td>
<td>Specify Full for the initial full data load, Incremental for a repeated incremental load, or Both to apply the parameter to the initial full data load and the repeated incremental load.</td>
</tr>
<tr>
<td>Value</td>
<td>Click this field to display the Enter Parameter Value dialog, where you specify a static value, runtime value, or SQL statement that returns a value. See What dynamic runtime variables are available?</td>
</tr>
<tr>
<td>Inactive</td>
<td>Select this field to deactivate a variable. For example, if you created a parameter for testing a data load, you might turn it off when you move to production. Before you deactivate a variable, make sure that it’s not being used. If you deactivate a parameter that is being used by a SQL command, the SQL command will fail if no override value is available.</td>
</tr>
</tbody>
</table>

What dynamic runtime variables are available?

For parameters of type Text, these variables are available.

<table>
<thead>
<tr>
<th>Variable name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>%</td>
<td>The current run’s process ID.</td>
</tr>
<tr>
<td>%CURRENT_PROCESS_ID</td>
<td></td>
</tr>
<tr>
<td>Variable name</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>%%LAST_PROCESS_ID</td>
<td>The last successful run's process ID.</td>
</tr>
<tr>
<td>%SOURCE_TABLE_OWNER</td>
<td>The dataflow's source connection's Schema Name/Table owner as defined by the user.</td>
</tr>
<tr>
<td>%TARGET_TABLE_OWNER</td>
<td>The dataflow's target connection's Schema Name/Table Owner as defined by the user.</td>
</tr>
<tr>
<td>%SOURCE_DBNAME</td>
<td>The dataflow's source connection name.</td>
</tr>
<tr>
<td>%TARGET_DB_NAME</td>
<td>The dataflow's target connection name.</td>
</tr>
<tr>
<td>%READ_MODE</td>
<td>The read mode indicating whether the data read is a full read or incremental read.</td>
</tr>
<tr>
<td>%WRITE_MODE</td>
<td>The write mode indicating whether the data is written by replacing data or incrementally applied (append or upsert mode).</td>
</tr>
</tbody>
</table>

For parameters of type **Timestamp**, these variables are available.

<table>
<thead>
<tr>
<th>Variable name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>%ETL_START_TIME</td>
<td>The local timestamp of when the job started.</td>
</tr>
<tr>
<td>%CURRENT_TIMESTAMP</td>
<td>The current local timestamp of when the parameter is evaluated just before the execution of a task.</td>
</tr>
<tr>
<td>%SOURCE_REFRESH_TIMESTAMP</td>
<td>The timestamp of the last successful job which touched the source table.</td>
</tr>
<tr>
<td>%TARGET_REFRESH_TIMESTAMP</td>
<td>The timestamp of the last successful job which touched the target table.</td>
</tr>
<tr>
<td>%SOURCE_PRUNED_REFRESH_TIMESTAMP</td>
<td>The last refresh timestamp of the source minus the prune time specified at the connectivity parameters of the job.</td>
</tr>
</tbody>
</table>

**Note:** When using generic JDBC data sources, you must choose custom format, and provide the date representation in the Java timestamp format. If you do not, Data Sync can’t evaluate the timestamp.

**Help: Patch Alerts Dialog**

To display a list of patches that have been applied to your Data Sync installation, click the **New patch alerts** icon in the top right hand corner of the Data Sync main screen.

**Help: Pluggable Data Sources Dialog**

The **Pluggable Source Data** tab enables you to configure data loads from many popular data source types, such as JDBC, and OTBI. We advise you not to change the installed sources Generic JDBC, Oracle BI Connector, or Oracle Service Cloud (RightNow).

Use the Pluggable Source Data tab:
Help: Pre/Post SQL Processing Dialog

This view enables you to edit your target Oracle Database Cloud Service data before or after a data load.

Note:

If you have a default Database Schema Service target with your Oracle BI Cloud Service, then you can’t use Data Sync to pre-process or post-process your data. You must have Oracle Database Cloud Service to use this functionality.

Edit Tab

The edit tab displays details of the post load processing operation that is currently selected.

<table>
<thead>
<tr>
<th>Field or element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;List of processes&gt;</td>
<td>View and edit processing operations for the current project. Click the SQL(s)/stored Procedure(s) field to display the SQL(s)/Stored Procedure(s) dialog and review SQL statements and functions, and add new logic. Use the Inactive option to activate or deactivate the logic. To create a new operation, click New, specify a name for the process, use the Pre/Post option to specify whether to execute the SQL before or after the data load, and use the SQL(s)/stored Procedure(s) option to display the SQL(s)/Stored Procedure(s) dialog, which enables you to define your SQL statements and functions. Notes about SQL Statements: • SQL statements and procedures execute one after the other in the order specified in the list. • Oracle recommends that you design SQL statements as re-entrant. If failures or restarts occur, then all statements are re-executed irrespective of where the prior failure happened. • In the Sql Statement box, you can specify parameters that are defined on the Parameters tab or Execution Parameters tab. To specify a parameter, either enter the parameter name prefixed with %%%, or expand the FUNCTIONS &amp; PARAMETERS/SOURCE SYSTEM PARAMETERS tree node and double–click a parameter name.</td>
</tr>
<tr>
<td>Name</td>
<td>Specify a short name to identify the SQL processing operations in the Data Sync client tool and in log files.</td>
</tr>
</tbody>
</table>
### SQL Source Tables Tab

Specifying source tables is optional. You specify source tables to identify the tables that SQL statements read from. When specified, as soon as the extraction of the source tables is complete, the specified SQL statements execute, even if there are other tables still being extracted. This helps to optimize the overall execution time. If you don’t specify source tables, Data Sync defers the step until all extraction tasks are complete in the current project.

<table>
<thead>
<tr>
<th>Field or element</th>
<th>Use to</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pre/Post</td>
<td>Choose <strong>Pre</strong> or <strong>Post</strong> to specify whether to execute the SQL code before data load (using <strong>Pre</strong>) or after data load (using <strong>Post</strong>).</td>
</tr>
<tr>
<td>Inactive</td>
<td>Activate or inactivate the process.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Field or element</th>
<th>Use to</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add/Remove</td>
<td>Display the Choose Tables dialog, where you specify the tables to include in your SQL processing operation.</td>
</tr>
<tr>
<td>Remove</td>
<td>Remove the selected table from the load processing setup.</td>
</tr>
</tbody>
</table>

### SQL Target Tables Tab

Specifying target tables is optional. You specify target tables to edit tables before SQL execution (for example, to create or alter tables), or perform analysis after SQL execution (for example, to analyze table statements). In addition, this tab enables you to specify when to truncate a table (similar to providing a load strategy).
<table>
<thead>
<tr>
<th>Field or element</th>
<th>Use to</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add/Remove</td>
<td>Display the Choose Tables dialog, where you specify the tables to include in your SQL processing operation.</td>
</tr>
<tr>
<td>Remove</td>
<td>Remove the selected table from the load processing setup.</td>
</tr>
<tr>
<td>Truncate Always</td>
<td>Use this option if a SQL statement creates an aggregate table with content that is fully refreshed.</td>
</tr>
<tr>
<td>Truncate For Full Load</td>
<td>Use this option if SQL statements append or updates existing data.</td>
</tr>
</tbody>
</table>

**Help: Project Summary Dialog**

The Project Summary tab enables you to manage your data load settings.

**Help: Properties Dialog**

You use properties to specify the data you want to load and how you load it.

For example, to set up a data load from a RightNow report, you specify the report ID of the data report and the report ID of the metadata report. You can review your property settings later under the Project, Pluggable Source Data, Pluggable Attributes tab.

**Choosing which Properties to Display**

Click the list and select from:

• Original — display the default list of Names and Values for the type of data load that you’ve selected.

**Specifying Values**

Click the Value field and use the displayed dialog to enter or copy in a value. For example, you might copy in a SQL statement or ROQL statement.

**Note:** READ_TYPE displays the query type that was selected when the Pluggable Data Source was created, and is read-only. If you want to change the query type, create a new Pluggable Data Source and select a different query type from the Data from option.

**Specifying Timestamps in RightNow Queries**

When you specify a timestamp in a query on an Oracle Service Cloud RightNow data source, the timestamp must be in the format:

```
yyyy-MM-ddTHH:mm:ssZ
```

For example, you might filter a query using:

`updatedtime > '2014-01-01T00:00:00Z`.`

**Using Partition Reads**

Specify a partition read when a query would otherwise load more records than the maximum fetch size allowed for your data source or target Cloud service, or as a workaround to memory issues.
Help: Relational Data Dialog

You can load data into your target Cloud database directly from either a relational table, a view, or a SQL statement.

Loading Data from Tables

You can import table definitions to load from using any of the defined relational connections. Supported data types include CHAR, VARCHAR, TIMESTAMP, DATE, NUMBER(n), NUMBER(m,n), CLOB, and BLOB. If a source table has columns with any other data type, they are imported with an UNKNOWN data type, and the column will be marked as inactive, and will not participate in the data copy process.

1. In the Project view, select the Relational Data tab.
2. Click Data From Table.
3. In the Import Tables into [Project] dialog box, select the connection in the Data Sources list.
4. In the Table Filter field, enter a table name or a table name filter, using wild cards to narrow the list of tables for import. The following examples filter the list of tables from a source.
   - CONTACT will show only the CONTACT table if it exists in the database with exactly the same name.
   - CONTACT* or CONTACT% lists all tables in the database whose name start with CONTACT.
   - *CONTACT* or %CONTACT% lists all tables in the database whose name contains CONTACT.
5. Click Search Tables.
6. In the Searching Tables confirmation dialog box, click OK. The Table List includes all tables from the source meeting the filter you applied, if any.
7. Select the Import check box for any tables in the list you want to replicate, then click Import Tables to register the source tables and create entries with the same name for target tables. All columns and indexes are also imported.

Note:

Do not rename the tables. Data Sync assumes that the source table name and target table name are the same. If you want to use a different target table name, consider using queries as a source.

8. If you chose to load data incrementally a unique index is suggested on the user/primary key columns. It is also recommended that you register additional indexes which support joining with other tables and can be used for filtering purposes while reporting.

By default, all table attributes are copied. If you want to preclude certain columns from being replicated to your target service because they are not needed for analysis or may contain sensitive information, select the table in the Target Tables tab, then select the Table Columns sub-tab and select the Inactive check box for the column or columns. If you deactivate a column, be sure to inspect the index definitions which
may be referring to the inactive columns. Any index that refers to an inactive or deleted column definition gets dropped, but is not created. If you would like to deactivate the indexes that may refer to inactive or deleted columns, right-click the column and select the identify and inactivate invalid indexes option. This marks any indexes that refer to inactive columns inactive as well.

Using a SQL Query Override to Refine a Data Load from a Table

You can limit data from a source table using a SQL query override.

By default, all data from a source table is loaded. To limit the data loaded, you can provide a SQL query override to refine the data that is read. For example, if you want to copy one year's worth of data on a table that includes a LAST_UPD date column, you could provide an additional query (Oracle Syntax) as:

```
SELECT * FROM TABLE_NAME WHERE LAST_UPD > SYSDATE - 365
```

1. In the Project view, select the source table in the Relational Data tab.
2. In the Edit sub-tab, click the button for the Query field.
3. In the Query dialog box, use the editor tools to enter your SQL query.
4. Click OK.

When you provide a SQL query override, Data Sync validates the SQL and prompts you to correct any errors. If the SQL override includes new columns that are not present in the table definition, you're prompted to add them to the target table.

For example, take a case where a CONTACT table is imported. By default, Data Sync issues `SELECT * FROM CONTACT`. You may want to add a column named UPLOADED_DT to the table to record when the data is uploaded. To do this, provide a SQL query override such as the following:

```
SELECT CONTACT.*, SYSDATE AS UPLOADED_DT FROM CONTACT
```

In this case, Data Sync recognizes that the new column UPLOADED_DT doesn't exist on the target and offers to add it to the table definition.

Loading Data from SQL

You can load data based on a SQL statement.

Another approach to load data into the Oracle Business Intelligence Cloud Service schema is to use a SQL statement whose results you want to persist. For example, instead of loading detail data, you may want to use an aggregate SQL to store compressed data on the cloud. This aggregate SQL may join multiple tables and use any of the SQL functions your database supports such as GROUP BY, filters, and joins.

1. In the Project view, select the Relational Data tab.
2. Click Data From SQL.
3. In the New Query dialog, enter a logical name for the query in the Name field. The name should not contain spaces.
4. Specify an existing target table or create a new one and provide a name for the table. If the query defines a new table, the column definitions are inferred from the SQL structure. If you use an existing table, any new columns from the SQL can be added to the list of columns.
5. Select a connection in the **Connection** list.
6. Enter the SQL query in the Query window.
7. Click **OK**.

If you chose to load data incrementally, a unique index is suggested on the user/ primary key columns. It is also recommended that you register additional indexes which support joining with other tables and can be used for filtering purposes while reporting.

### Help: System Properties Dialog

System Properties enable you to customize your Data Sync environment. To review system properties, select System Properties from the Views menu. To change a system property, click the **Value** column and enter your changes.

<table>
<thead>
<tr>
<th>Property</th>
<th>Use to</th>
</tr>
</thead>
<tbody>
<tr>
<td>Allow Clients To Remember User Password</td>
<td>Specify whether you can start Data Sync without specifying a repository password each time you log in. When set to <strong>true</strong>, you can start Data Sync without entering a password if you selected the Remember password option on the previous login. When set to <strong>false</strong>, you must always enter a password, even if you selected the Remember password option the last time you logged in.</td>
</tr>
<tr>
<td>Automatically Create/Alter Tables</td>
<td>Specify whether Data Sync creates targets tables for you when data is loaded. We recommend that you use the default value (true).</td>
</tr>
<tr>
<td>Concurrency Level</td>
<td>Specify the maximum number of jobs that can run in parallel. When concurrency of 1 is set, ETL runs in the same process space with the Data Sync. When multiple jobs are allowed, they run as separate operating system processes. Any invalid value is interpreted as 1. (Restart the Data Sync server to deploy changes.)</td>
</tr>
<tr>
<td>Data File Root Directory</td>
<td>Specify a default directory for importing data files. When set, the directory defaults to this location when registering a file to be uploaded.</td>
</tr>
<tr>
<td>Delete Data Cache</td>
<td>Specify whether to delete the data cache for data model objects. When set to <strong>true</strong> (the default), the cache is deleted at the end of every job. When set to <strong>false</strong>, the cache is not deleted. To load data to Database As A Service, when you create a data loading job, you must also use the <strong>Cloud Connection for Cache Purging</strong> option on the Jobs\Edit tab to specify which cache to purge.</td>
</tr>
<tr>
<td>Heartbeat Interval</td>
<td>Specify how often (in seconds) Data Sync performs consistency and diagnostics checks. Don't change the default value of 900 seconds (15 minutes) unless advised by Oracle Support. Specifying more frequent checks can negatively affect performance.</td>
</tr>
<tr>
<td>Maximum number of run history entries to preserve</td>
<td>Specify the maximum number of run history entries to preserve. Older entries in the run history are purged. A value less than or equal to zero preserves everything. This is not recommended for Data Sync, where the repository is maintained in the Java database.</td>
</tr>
<tr>
<td>Proxy Host/Proxy Port</td>
<td>Specify details of your proxy server, if you have one. If you don’t have a proxy server, then leave these values empty. (Restart the Data Sync server to deploy changes.)</td>
</tr>
</tbody>
</table>
### Help: Target Option Dialog

This dialog enables you to select an existing target table into which you load your file-based data.

<table>
<thead>
<tr>
<th>Element or field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;List of targets&gt;</td>
<td>A list of existing target tables into which you can load your file-based data. For example, if you have data in multiple files you might want to load them into the same target table for analysis by your BI users. Select a target table then click OK.</td>
</tr>
</tbody>
</table>

### Help: Target Tables and Data Sets Dialog

The **Target Tables/Data Sets** tab enables you to configure metadata in the target cloud area.

### Help: Welcome Dialog

Use this dialog to start working in an existing data loading project or create a new data loading project.

You use projects to manage your data loads. For example, if you want to load data from two separate data sources, Data1 and Data2, you typically create a separate project for each data source. When you create or select a project, you'll be working in that project when you're in the Project or Jobs view.

To create a new data loading project from the main Data Sync workarea, select File, then Projects to display this dialog and select **Create a New Project**. To open an
existing data loading project from the main Data Sync workarea, use the option to the left of the Run Job option.

<table>
<thead>
<tr>
<th>Field or element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create a New Project</td>
<td>Create a new data loading project using a descriptive name (30 characters maximum) that you specify in the box below this option.</td>
</tr>
<tr>
<td>Select an Existing Project</td>
<td>If you have previously created a data-loading project, select it in the list below. If you don’t know which existing project you want to work in, select Skip Create/Select a Project, and use the option to the left of the Run Job option to navigate to different projects.</td>
</tr>
<tr>
<td>Skip Create/Select a Project</td>
<td>Start up Data Sync without selecting a data project, for example, you might want to create some connections to a data source and a data target. In the main Data Sync work area, you can also select a project to work on in the Project or Jobs view, using the list to the left of the Run Job option.</td>
</tr>
</tbody>
</table>

Help: Clearing the Cache After Uploading Data

By default, the data cache is deleted after each job run.

By default, the data cache for data model objects is deleted at the end of every completed job run. To support this functionality, grant the user registered to upload data the BI Data Modeler role.

Jobs run to completion regardless of whether the attempt to purge the cache succeeds.

Note:

Upon job completion, log off and log back on to see the effect of the purged cache. Changes may take several minutes.

If you do not wish to have the cache deleted, set the Delete Data Cache system property to false.

Help: Creating and Modifying Tables

During and prior to data load to a Cloud Service target, tables are created automatically and can be manually created.

When you load data, target tables are created on your Cloud schema automatically prior to loading the data. For subsequent data load runs, the table definition from your local repository is compared against that in your Cloud schema, and altered dynamically. You can also create the tables prior to load manually by navigating to the Target Tables tab in the Project view, right-clicking, choosing Drop/Create/Alter Tables, and selecting your target. This process causes errors if the table is not alterable. Examples of such situations include:

- Changing a null column into a not null column.
- Changing the data-type.
- Reducing the length or precision.
If errors are reported, you can manually drop and re-create the table by right-clicking the target table in the Target Tables list in the Project view and selecting Drop/Create/Alter tables.

**Note:**
Dropping and re-creating the table clears the refresh date for that table. This will force a full load on the table in the subsequent load.

Creating and Modifying Other Types of Table

For trial run purposes or testing, you might use Data Sync to load data to somewhere other than your target Cloud database, such as an on-premises database. In this case, you should create and alter the tables manually prior to data load. Navigate to the Target Tables tab in the Project view, right-click, and select Drop/Create/Alter Tables.

About System Columns

As Data Sync streams the data to your Cloud service, communication-related failures can occur. To address this scenario, Data Sync automatically retries 10 times before reporting a failure. In the case of an insert/append scenario, in order to prevent data duplication, retries within a streaming process or across job restarts require data clean-up before each attempt. For every table that gets loaded, Data Sync adds the following system columns in order to enable this functionality:

- DSYS_INSTANCE_ID: Tracks the Data Sync installation instance ID.
- DSYS_PROCESS_ID: Tracks the process ID assigned to a certain run of the job.
- DSYS_BATCH_ID: Tracks the batch that is trying to upload the data. Each table load streams multiple batches (currently of 3,000 rows), with each batch assigned a unique number.

About Rolling Deletes

If you wish to load a subset of your data for the past 'n' number of days, you can control this behavior by providing a SQL Query Override. However, as time passes, the data in the Cloud schema continues to accumulate. If you want to limit data in the Cloud schema to a certain period and periodically purge older data, you can specify Rolling Delete Days at the target table level. For example, for the CONTACT table, if you want to purge all data greater than a year, in the Target Tables tab in the Project view, enter 365 for Rolling Delete Days for the table. You can set this in the table list or in the Edit sub-tab.

It is important to define an index on the column used to identify the older records. This improves data load performance.

Help: Consolidating Data from Multiple Sources

In the case that you have different kinds of sources in your environment, you may want to consolidate their data for analytic purposes.

Multiple sources can be of three types: heterogeneous, homogenous (of the same version), and homogenous (of different versions).
Heterogeneous Sources

Heterogeneous sources are of different types, for example CRM, Oracle EBS, Peoplesoft, JDEdwards, and so on. The sources typically contain mutually exclusive sets of tables. You can either choose to use the same project or different ones. A single project is convenient when you have a single schedule to upload the data. Multiple projects, one per source are convenient when you need to schedule dataload at different times and/or multiple people work on them.

Homogenous Sources of the Same Version

Homogenous sources of the same version occur when you have the same kind of source, but multiple instances of it. For example, you could have two instances of CRM, one used for North America and the other for Europe. You can manage data extraction from both of these sources with a single project.

To manage data extraction for homogenous sources of the same version in the example described, consider the following tips and requirements:

• Create a connection for the database in North America and another for the one used for Europe.
• Use one of the databases for identifying the tables and queries to replicate.
• Test the load process from one source.
• Create a new job, where you can map the existing source (with which you defined metadata) to the new one.
• Schedule the jobs separately.

Homogenous Sources of Different Versions

Homogenous sources of different versions are very similar to the case of homogenous sources of the same version. The only difference is that there are tables that may have the same name but different structures and schema definitions. There are two ways of performing data upload: using a single project or using multiple projects, one per type of source.

To use a single project:

• Use one source as a primary source to import the table definitions.
• Use Query as a means of importing data from the others.
• If there are new attributes from the queries, add them to the target table definition.

To use multiple projects:

• Create individual projects, one per version of the source system.
• Import tables and queries from sources into their respective projects.
• Take care that the target tables to have similar datatypes. For example, if COLUMN1 for Table1 in source1 is of type VARCHAR, and the same table column has a type of DATE in the other, the data loads from one or the other source will fail.

Help: Creating and Modifying Data Sets

You can use Data Sync to load your data as data sets.
• Data Sync creates the target data set automatically with default settings. String and timestamp based data is characterized as attributes, and numeric data is characterized as measures.

• You can modify data sets. Subsequent data loads preserve customizations.

• You can add more attributes to your data set in subsequent data loads. These attributes are created in the data set but are not enabled.

• If a subsequent data load changes the datatype of any attribute, or removes an existing attribute, then Data Sync reports an error. To correct this error, delete the data set and re-create it (in the Project view, display the Target Tables/Data Sets tab, right-click the table and click Drop/Create/Alter Tables/Data Sets).

• The maximum data set size is 50MB. Data uploads fail if the data set exceeds the 50MB limit.

Help: Triggering Jobs from Other Tools

In some cases, you might want to trigger a data load from an extraneous process.

There are three ways of integrating with other processes: file-based, command line based, and SQL-based.

File-Based Integration

Edit the on_demand_job.xml file located in the conf-shared directory, and specify a file that will trigger a specified job. The TriggerFile job parameter enables you to specify a job and a file whose existence triggers the job. For example:

Create an empty text file named abc.txt in the conf-shared directory. Then edit the on_demand_job.xml file and set the TriggerFile parameter to:

<TriggerFile job="Job1" file="c:\abc.txt"/>

In this example, Data Sync polls for the presence of the abc.txt file, and when found it triggers Job1 execution and then deletes the abc.txt file.

You can also specify time windows for polling, as in this example in which file polling happens between the 12:00 AM and 2:30 AM and between 6:00PM and 8:00PM every day:

<TriggerFile job="Job2" file="c:\xyz.txt">
<TimeWindow startTime="00:00" endTime="02:30"/>
<TimeWindow startTime="19:00" endTime="20:00"/>
</TriggerFile>

Command Line Based Integration

Use the datasyncCmdLine.bat/.sh file to start a job and to obtain the status of a running job. The instructions for using this file are contained within the file as REM comments. The datasyncCmdLine file works in conjunction with the dac.properties file.

SQL-Based Integration

Use SQL-based integration when an external process needs to dictate when a job can begin, if it can perform a SQL operation on a data source. To integrate using this method, you create a trigger table that can be polled to initiate a job.
1. In the Connections view, create a connection.

2. Navigate to the conf-shared directory and edit the on_demand_job.xml file. Edit the following properties:
   - The polling interval.
   - The datasource name that has the trigger table.
   - The time periods between which Data Sync should poll the table.

3. Create a table on that data source called JOB_TRIGGER containing the following four columns:

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Datatype</th>
<th>Length</th>
<th>Values</th>
<th>Unique</th>
</tr>
</thead>
<tbody>
<tr>
<td>UNIQUE_KEY</td>
<td>VARCHAR</td>
<td>250</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>JOB_NAME</td>
<td>VARCHAR</td>
<td>250</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>EFFECTIVE_DT</td>
<td>DATE</td>
<td></td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>INACTIVE_FLG</td>
<td>CHAR</td>
<td>1</td>
<td>Y/N</td>
<td>No</td>
</tr>
</tbody>
</table>

Example of the Data Definition Language for a trigger table and index:

```sql
CREATE TABLE JOB_TRIGGER
(
  UNIQUE_KEY VARCHAR(250) NOT NULL,
  JOB_NAME VARCHAR(250) NOT NULL,
  EFFECTIVE_DT DATE NOT NULL,
  INACTIVE_FLG CHAR(1) DEFAULT 'N' NOT NULL
)
;

CREATE INDEX JOB_TRIGGER_U1 ON JOB_TRIGGER(UNIQUE_KEY)
;
```

An entry in this table now triggers the job upload. As soon as the job request is started, the entry is deleted. You can also specify during which period polling should occur.

**Help: Triggering One Job After Another Automatically**

In some situations, you may want a job triggered upon completion of another job.

Common scenarios in which you may want a job triggered upon completion of another job include but are not limited to the following:

- If there are multiple jobs writing to the same target, you want to stagger them because there is a governance that limits the number of dataloads that can occur for a connection.
- By default, within a project, all the tables, SQL queries, and file data get loaded in parallel, with no specific order other than staggering dataflows that write to the same table. If you would like the tables to be staggered, for example to load details first and then summaries, then you can create two separate projects, one for the detail tables and the other for the parent tables. When the job that loads the details is complete, you want to trigger the parent summary loads.

To enable this, when the job starts a signal file with a naming pattern `<JOB_NAME_WITH_NO_SPACES>_StartSignal.txt` is created in the log\jobSignal directory for each run of the job. A file with the naming pattern `<JOB_NAME_WITH_NO_SPACES>_CompletedSignal.txt` is created when the job
completes successfully, or <JOB_NAME_WITH_NO_SPACES>_FailedSignal.txt when the job fails. These files can be used with the Data Sync's file poll mechanism to chain jobs to meet your needs. For more information about the file poll mechanism, see Integrating With Other Processes.

Help: Uploading Data to Multiple Cloud Targets

If you have multiple cloud targets, for example, development and production, then you load data to each target. There are two ways to do this: you can use the same environment, but with multiple data load jobs, or you can set up a mirror environment where you can import the data from the development environment.

1. Determine the source and, if necessary, create a new data source in the Data Sync client.
   When you set up data sources, there are two possibilities for the source database:
   • You want to extract from the same source and populate a production target on the cloud.
   • You want to extract from a different source and populate a production target. In this case, create another data source that points to the source you want to populate the production environment from.

2. For the target database, create another connection of type "Oracle (Oracle Business Intelligence Cloud Service)".
   For example, you might name it Production.

3. In the Jobs tab in the Jobs view, create a new job.
   In the New Job dialog box, specify a meaningful name, for example, Production Job, then click Next.

4. The next page of the New Job dialog box displays the currently used connections.
   To remap these to the new source (if applicable) and the newly created target connection, select the new connections in the Override With column, then click Finish.

Setting Up a Different Environment

Setting up a different environment is preferable when there are a lot of updates happening on the development schema, or when the teams responsible for the development and production environments are different. To set up a different environment for the first time, export the system and its logical metadata from the development environment and import it into the production environment by choosing to truncate the tables using the Export and Import commands on the Tools menu. After initial setup, export only the logical metadata from the development environment and import into production by choosing to truncate the tables.
Help: Column Mapping/Mapping Dialog

You use this dialog to review how source columns are mapped to target columns, and to transform your relational data. For example, you might convert values to uppercase, create calculations, or create lookups.

Column Mapping/Mapping Dialog

You typically use this dialog to configure data transformations.

<table>
<thead>
<tr>
<th>Field or Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Joins</td>
<td>Display the Joins dialog, where you can create lookups and denormalize data. See Creating Joins below.</td>
</tr>
<tr>
<td>Unmapped Columns</td>
<td>Display the Choose Columns dialog, which enables you to add new columns to your target database. For example, if you click New and create a new target column that doesn’t exist in the data source, click Unmapped Columns and move the new column to the Selected Columns list.</td>
</tr>
<tr>
<td>New</td>
<td>Create a new column. For example, you might want to calculate Return on Investment and store the value in a new column named ROI with the Target Expression defined as (REVENUE * (DISCNT_RATE/100)) – COST.</td>
</tr>
<tr>
<td>Source Column Name</td>
<td>The column name in the data source, or defined when the column was created.</td>
</tr>
<tr>
<td>Source Column Type</td>
<td>The column type in the data source, or defined when the column was created.</td>
</tr>
<tr>
<td>Data Transformation</td>
<td>Apply simple transformations to target columns. For example, you might convert text to uppercase, or use the FILE_NAME option to track where data originates. Alternatively, you can transform data using any supported SQL expression in the Target Expression field.</td>
</tr>
<tr>
<td>Target Column Name</td>
<td>The column name in the target database, typically defaulted to the Source Column Name.</td>
</tr>
<tr>
<td>Field or Element</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Target Expression</td>
<td>The SQL expression that performs the data transformation. Click here to display the Expression editor, which enables you to build a SQL expression to transform your data. On the Expression editor, enter column names or click column names in the left-hand pane to copy them into the expression. Note that tables named with %% are staging tables. To transform relational data:</td>
</tr>
<tr>
<td></td>
<td>• To specify a default value, click the <strong>Target Expression</strong> field to display the Expression dialog, and use the <strong>Default</strong> option to enter a default value, or select one of the installed values (for example, %UPsert_TIMESTAMP).</td>
</tr>
<tr>
<td></td>
<td>• To create a surrogate key, on the Column Mappings dialog click <strong>New</strong>, specify the details, click the <strong>Target Expression</strong> field, and on the Expression dialog select %SURROGATE_KEY in the <strong>Default</strong> option.</td>
</tr>
<tr>
<td></td>
<td>• To calculate or update the target value, on the Column Mappings dialog click <strong>New</strong>, specify the details, click the <strong>Target Expression</strong> field, and on the Expression dialog use the Expression option to specify the calculation. For example, if you want to specify a COST value as '0' if it’s less than '0' and assign a default '0' if no value is available in the data source, then specify CASE WHEN COST &lt; 0 THEN 0 ELSE COST END in the <strong>Expression</strong> field and enter 0 in the <strong>Default</strong> field.</td>
</tr>
<tr>
<td></td>
<td>• To create a new target column, on the Column Mappings dialog click <strong>New</strong>, and specify the target column details. Click <strong>Target Expression</strong>, and on the Expression dialog use the Expression option to specify the calculation. For example, if you want to create a Return On Investment value based on Cost and Discount Rate, then you might specify (REVENUE * (DISCNT_RATE/100)) - COST in the <strong>Expression</strong> field and enter 0 in the <strong>Default</strong> field. Then click <strong>Unmapped Columns</strong>, and on the Choose Columns dialog add the new column to the <strong>Selected Columns</strong> field.</td>
</tr>
<tr>
<td></td>
<td>• To concatenate data, on the Column Mappings dialog click the <strong>Target Expression</strong> field for the target column, and on the Expression dialog use the Expression option to specify the concatenation expression. For example, if you want to concatenate a FULL NAME field, then you might specify last_name</td>
</tr>
<tr>
<td></td>
<td>• To add runtime values to the target data, on the Column Mappings dialog click the <strong>Target</strong></td>
</tr>
</tbody>
</table>
Creating Joins

You can use joins to denormalize data, and perform data lookups. Click Joins to display the Joins dialog, which enables you to manage your lookups and joins.

<table>
<thead>
<tr>
<th>Field or Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;List of joins&gt;</code></td>
<td>A list of existing joins that are available to use in the currently selected project.</td>
</tr>
<tr>
<td>Field or element</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>New</td>
<td>To create a new join, click <strong>New</strong>, and specify the following details:</td>
</tr>
<tr>
<td></td>
<td>• <strong>Name.</strong> Specify a short user-friendly name to identify the join in Data Sync. For example, LOOKUP_CUSTOMER.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Table Names.</strong> Click this field to display the Table Names dialog, where you specify the names of the tables to join, separated by commas. For example, PRODUCT, ORDERS.</td>
</tr>
</tbody>
</table>
|                  | • **Join.** Click this field to display the Join dialog, where you build your join SQL statement. On the Join editor, enter table or column names or click table or column names in the left-hand pane to copy them into the expression. Note that tables named with % are staging tables. You can join more than one table in a join statement (in the ansi sql style). You can also define aliases for the tables you're joining, using `alias.columnName`. The base table is a runtime stage table, therefore you should refer to it as the target table name prepended with %.

For example, to load ORDER table with a join to PRODUCT table, specify:

```
INNER JOIN PRODUCT ON %%ORDER.PRODUCT_ID = PRODUCT.PRODUCT_ID
```

Or:

```
LEFT OUTER JOIN PRODUCT ON %%ORDER.PRODUCT_ID = PRODUCT.PRODUCT_ID
```

|                  | **Note:** |
|                  | If a join is expected to have one match, use an inner join. If a join is expected to have multiple matches, use an outer join. |
|                  | • **Yields Multiple Matches.** Click this option if a join is expected to return multiple matches. If it yields more than one possible match, then use an aggregate function that refers to a column from this join statement. |
|                  | • **Inactive.** Deactivate or activate the join. |