Oracle® Cloud

Administering Oracle NetSuite Analytics Warehouse

F35237-38
March 2025

Oracle Cloud Administering Oracle NetSuite Analytics Warehouse,

F35237-38

Primary Author: Padma Rao

Contributing Authors: Jenny Smalling

Contributors: Oracle NetSuite Analytics Warehouse development, product management(contributors), and quality assurance teams

Contents

1 Get Started with Oracle NetSuite Analytics Warehouse

2 Configure Oracle NetSuite Analytics Warehouse Data

3 Manage Users, Groups, and Application Roles

6 Integrate Third-Party Data

Preface

Learn how to get started with Oracle NetSuite Analytics Warehouse.

Topics

Audience

This document is intended for the administrators and advanced users of Oracle NetSuite Analytics Warehouse.

Related Documentation

These related Oracle resources provide more information.

- Oracle Cloud http://cloud.oracle.com
- Using Oracle Fusion Data Intelligence
- Getting Started with Oracle Analytics Cloud
- Visualizing Data and Building Reports in Oracle Analytics Cloud
- Configuring Oracle Analytics Cloud

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

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.

Conventions

The following text conventions are used in this document.

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

1

Get Started with Oracle NetSuite Analytics Warehouse

Let's explore Oracle NetSuite Analytics Warehouse and what you need to know to get started.

Topics:

About Oracle NetSuite Analytics Warehouse

Oracle NetSuite Analytics Warehouse enables you to analyze your NetSuite data and generate visualizations and reports.

This service is based on the real-world notions of KPIs, cards, and decks, and provides a collaborative experience optimized for executives and decision-makers. Business analysts can extend Oracle NetSuite Analytics Warehouse using the intuitive KPI editor and other Oracle Analytics Cloud features to author new visualizations, reports, and dashboards. These analytics, whether new or ready-to-use, work alongside KPIs, cards, and decks for a comprehensive data analysis experience.

This packaged service includes Oracle Analytics Cloud and is powered by Oracle Autonomous Data Warehouse. The service extracts and loads data from your Oracle NetSuite instance into an instance of Oracle Autonomous Data Warehouse. You can then use Oracle Analytics Cloud to customize existing dashboards and create dashboards. You can augment your NetSuite data with data from the custom transaction objects and make the custom data available for reporting.

Oracle NetSuite Analytics Warehouse consists of a data pipeline, data warehouse, semantic model, and best-practice content such as ready-to-use KPIs, dashboards, and reports. Oracle manages the service starting with deployment through performance monitoring, upgrades, and maintenance for the ready-to-use content.

Oracle NetSuite Analytics Warehouse supports several non-English languages, and certain elements such as the user interface and auto-generated text are displayed in those languages.

You can extend your Oracle NetSuite Analytics Warehouse subscription by buying additional user packs. Each user pack contains licenses for five users. User packs available for these license tiers are:

- Standard tier: seven user packs
- Premium tier: nine user packs
- Enterprise tier: No limit on user packs

For example, if you have a Standard Oracle NetSuite Analytics Warehouse license, you can only buy up to seven user packs. If you require nine user packs, then you must upgrade to the Premium license tier.

Region Availability

Oracle NetSuite Analytics Warehouse is available in the following regions currently:

Region Location	Region Name	Region Key
Ashburn, VA	us-ashburn-1	IAD
Frankfurt, Germany	eu-frankfurt-1	FRA
London, United Kingdom	uk-london-1	LHR
Phoenix, AZ	us-phoenix-1	PHX
Sydney, Australia	ap-sydney-1	SYD
Tokyo, Japan	ap-tokyo-1	NRT

Language Support

Oracle NetSuite Analytics Warehouse is supports the following non-English languages:

- Arabic
- Chinese Simplified
- Chinese Traditional
- Dutch
- French Canadian
- French European
- German
- Italian
- Japanese
- Portuguese Brazilian
- Spanish

About the Autonomous Data Warehouse Tools

If you want to use the tools available in Autonomous Data Warehouse associated with your Oracle NetSuite Analytics Warehouse instance, then note the following:

- If you’re an Oracle NetSuite Analytics Warehouse (Standard and Premium) customer, then the Database tools option for only "Data Transforms" is enabled.

- If you're an Oracle NetSuite Analytics Warehouse (Enterprise) customer, then the Database tools option for only Oracle REST Data Services (ORDS), Oracle Application Express (APEX), Data Transforms, and Oracle Machine Learning is enabled.
- If you wish to use the Database tools that are enabled, then follow this process:
  - Download the wallet and open the README file. See About Autonomous Data Warehouse Wallet and Administrator Credentials.
  - In the README file, copy any url for the Database tools listed in the Autonomous Database Tools and Resources section till `.com` and then append the url after `.com` for Data Transforms with `odi/oracle-data-transforms/` and for Oracle REST Data Services (ORDS) with `/ords/`. You can use the URLs for Oracle Application Express and Oracle Machine Learning that are available in the README file.

Enable Single Sign-On

Set up the single sign-on authentication to enable users to access their Netsuite account through Oracle NetSuite Analytics Warehouse.

When you enable single sign-on in Oracle NetSuite Analytics Warehouse, you allow single sign-on access to NetSuite using authentication from a third-party identity provider. Users who have signed in to Oracle NetSuite Analytics Warehouse can directly go to NetSuite. They don't need to log in separately to NetSuite, because authentication from the same identity provider is used for login to both Oracle NetSuite Analytics Warehouse and NetSuite. To complete the single sign-on integration with a third party identity provider such as OKTA, Google, or Azure, you must refer to the applicable third party identity provider's documentation on how to obtain the required values.

Ensure that you've configured SAML or OIDC single sign-on in NetSuite based on your buisiness usage. See Using Single Sign-On with NetSuite Analytics Warehouse.

1. In Oracle NetSuite Analytics Warehouse, open the **Navigator** menu, click **Console**, and then click **Authentication**.
2. On the Authentication page, select **Enable SSO**.
3. Under Authentication System Settings, in **Assertion Consumer URL**, enter the URL to direct your identity provider to send its SAML Response after authenticating a user.
4. In **Entity ID**, enter the name of your application in the identity provider.
5. In **NamedID Format**, ensure **Email Address** is selected and in **NamedID Value**, ensure that **Primary Email** is selected by default.
6. Expand **Advanced Settings**, select **Enable Single Logout**, and provide appropriate values in **Logout Binding**, **Single Logout URL**, and **Logout Response URL** to configure the logout behavior.

Preview Features

Oracle NetSuite Analytics Warehouse offers certain functionality as preview features for you to try. Preview features are high-confidence features that are complete from design, specification, and implementation perspective. These features have a roadmap commitment but may change before they become generally available.

You can evaluate the Preview features for production use as-is based on your specific requirements. If you've questions on the feasibility of deploying a specific Preview feature for production use in your instance, contact Oracle Support.

Preview features are disabled by default but administrators can enable them for use. See Make Preview Features Available.

Connectors

See Third Party Data Connectors

Functional Areas

Feature	Description
Order to cash	This feature provides you the ability to perform cross-transactional analyses with the Order to Cash (O2C) flow. After enabling this functional area, ensure you create a data pipeline and activate it. See Create a Data Pipeline for a Functional Area.
Procure to pay	This feature provides you with the ability to perform cross-transactional analyses with the Procure to Pay (P2P) flow. After enabling this functional area, ensure you create a data pipeline and activate it.

Pipeline Features

Feature	Description
Data Refresh Estimate	This feature enables prediction of completion time for data refresh. You can see the predicted time under Data Pipeline in the Pipeline Parameters section on the Pipeline Settings page. See About Data Refresh Performance. • **On** – Display the **Estimated Refresh Completion** detail under Data Pipeline in the Pipeline Parameters section on the Pipeline Settings page. • **Off** – Hide the **Estimated Refresh Completion** detail under Data Pipeline in the Pipeline Parameters section on the Pipeline Settings page. Default: Off Restart Required: No
Frequent Refresh Tables	This feature enables custom warehouse tables for frequent data refresh. See Schedule Frequent Refreshes of Warehouse Tables (Preview). • **On** – Display the **Frequent Refresh Tables** tab on the Pipeline Settings page. • **Off** – Hide the **Frequent Refresh Tables** tab on the Pipeline Settings page. Default: Off Restart Required: No
Functional Area Schedule Override	This feature enables you to override the schedules for data pipelines by specifying a different refresh schedule. See Override Data Pipeline Schedules for Functional Areas (Preview). • **On** – Display the **Functional Area Schedule Override** tab on the Pipeline Settings page. • **Off** – Hide the **Functional Area Schedule Override** tab on the Pipeline Settings page. Default: Off Restart Required: No

Feature	Description
Prioritized Data Refresh	This feature enables you to configure the warehouse tables for prioritized refresh ahead of other data in the regularly scheduled daily refresh cycles. See Prioritize Datasets for Incremental Refresh (Preview). • **On** – Display the **Prioritized Data Refresh** tab on the Pipeline Settings page. • **Off** – Hide the **Prioritized Data Refresh** tab on the Pipeline Settings page. Default: Off Restart Required: No

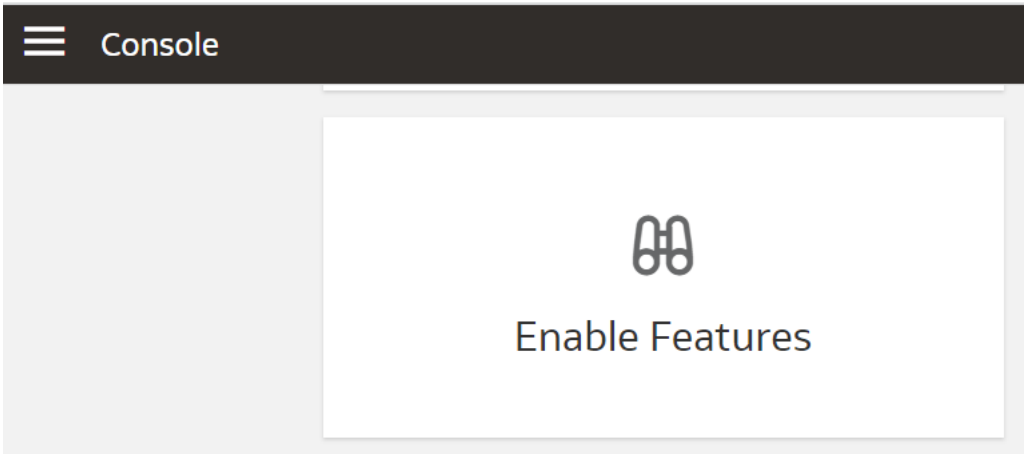
Make Preview Features Available

As a functional administrator, you can enable the functionality available as a preview feature to try it out.

Preview features allow your organization to explore and try new features before they roll out by default. See Preview Features.

The preview features aren't available by default, but administrators can turn individual preview features on or off at any time. Administrators can find the latest preview features in the Enable Features tile on the Console and switch them on for others to use.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Enable Features** under **Application Administration**.



3. On the Enable Features page, under the Preview Features tab, expand the categories such as **Functional Areas** and select the features that you want to make available for your organization.
4. Navigate to the location of the preview feature you want to use. For example, if you've enabled **AWS S3** in the **Connectors** category, then navigate to the Create Connections dialog from the Manage Connections page to use this feature.

Generally Available Features

Oracle NetSuite Analytics Warehouse offers certain functionality as generally available that you must enable using the Console. Generally available features are publicly available features that you can use for production workloads.

Generally available features are disabled by default but the administrators can make these features available for use; see Enable Generally Available Features.

Pipeline Features

Feature	Description
Frequent Refresh	This feature provides an ability to configure the Frequent Data Refresh process. Ensure to configure the frequent data refresh on the Pipeline Settings page by selecting the frequency and functional areas. This feature enables certain pipeline refresh schedules to be set to a frequency of 4 hours or more. See Schedule Frequent Refreshes of Data. • **On** – Display the **Frequent Refresh** option on the Pipeline Settings page. • **Off** – Hide the **Frequent Refresh** option on the Pipeline Settings page. Default: Off Restart Required: No
Scheduled Full Data Reload	For select functional areas, an administrator can schedule a full refresh on a weekly, monthly, or quarterly basis. This feature requires incremental data refresh to be scheduled for a daily frequency. See Schedule Periodic Full Reload of Functional Area Data. • **On** – Display the **Scheduled Full Data Reload** on the Pipeline Settings page. • **Off** – Hide the **Scheduled Full Data Reload** on the Pipeline Settings page. Default: Off Restart Required: No
Reset and Reload the Data Source	This feature enables reset and reload of a data source. You can reset and reload data for all the activated functional areas, augmentations, and custom data configurations. See Reset and Reload the Data Source. • **On** – Display the **Data Source Reset and Reload** detail under Data Pipeline in the Pipeline Parameters section on the Pipeline Settings page. • **Off** – Hide the **Data Source Reset and Reload** detail under Data Pipeline in the Pipeline Parameters section on the Pipeline Settings page. Default: Off Restart Required: No

Feature	Description
Reset Data Warehouse	This feature enables the option to reset the data warehouse (hard reset) and delete all data. The action is irreversible. During the data warehouse reset, all the activated functional areas are moved to “Saved” state and you must reactivate them. See Reset the Data Warehouse. • **On** – Display the **Reset Data Warehouse** action on the Pipeline Settings page. • **Off** – Hide the **Reset Data Warehouse** action on the Pipeline Settings page. Default: Off Restart Required: No
SME Options for Data Augmentation	This feature enables the semantic modeling options of Dimension, Extend Entity, and Fact in the Data Augmentation wizard. Augmentations that are configured as dimensions, custom fact, or extend entity won't be available for semantic model extensions. See Augment Your Data. • **On** – Display the **Create Dimension**, **Create Fact**, and **Extend Entity** in the **Augumentation Type** list of values in the Data Augmentation wizard. • **Off** – Hide the **Create Dimension**, **Create Fact**, and **Extend Entity** in the **Augumentation Type** list of values in the Data Augmentation wizard. Default: Off Restart Required: No

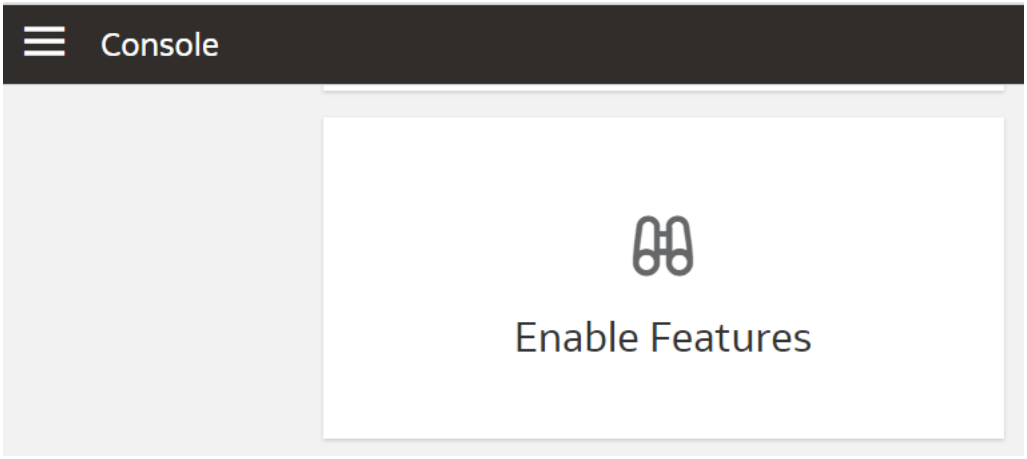
Enable Generally Available Features

As a functional administrator, you can enable the functionality that's generally available but needs you to enable it to use the specific functionality.

The generally available features aren't available by default, but administrators can turn individual features on or off at any time. Administrators can find the latest generally available features in the Enable Features tile on the Console and switch them on for others to use.

To find out about features that are generally available currently, see Generally Available Features.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Enable Features** under **Application Administration**.



3. On the Enable Features page, under the Generally Available Features tab, expand the categories such as **Pipeline Features** and select the features that you want to make available for your organization.

4. Navigate to the location of the feature you want to use. For example, if you've enabled **Frequent Refresh** in the **Pipeline Features** category, then navigate to the Pipeline Settings page and click Frequent Data Refresh Schedule to use this feature.

Third Party Data Connectors

Oracle NetSuite Analytics Warehouse enables you to connect to a variety of data sources and remote applications to provide the background information for reports. Certain data connectors are available as preview features while some are available by default.

Data Connectors Available as Preview Features

Feature	Description
AWS S3	This feature enables an administrator to connect to data in Amazon Simple Storage Service (AWS S3). See Load Data from Amazon Simple Storage Service into Oracle NetSuite Analytics Warehouse (Preview). • **On** – Display the **AWS S3** option in the Create Connection dialog. • **Off** – Hide the **AWS S3** option in the Create Connection dialog. Default: Off Restart Required: No
Azure SQL	This feature enables connecting to an Azure SQL instance to create augmentations using data from that instance. See Load Data from Azure SQL into Oracle NetSuite Analytics Warehouse (Preview). • **On** – Display the **Azure SQL** option in the Create Connection dialog. • **Off** – Hide the **Azure SQL** option in the Create Connection dialog. Default: Off Restart Required: No

Feature	Description
Azure Storage	This feature enables connecting to an Azure Storage instance to create augmentations using data from that instance. See Load Data from Azure Storage into Oracle NetSuite Analytics Warehouse (Preview). • **On** – Display the **Azure Storage** option in the Create Connection dialog. • **Off** – Hide the **Azure Storage** option in the Create Connection dialog. Default: Off Restart Required: No
Oracle Autonomous Database	This feature enables an administrator to connect to custom data in Oracle Autonomous Database. You can create up to five connections to different Oracle Autonomous Database instances and they are denoted as **Oracle Autonomous Database 1**, **Oracle Autonomous Database 2**, **Oracle Autonomous Database 3**, **Oracle Autonomous Database 4**, and **Oracle Autonomous Database 5**. See Load Data from Oracle Autonomous Database into Oracle NetSuite Analytics Warehouse (Preview). • **On** – Display the Oracle Autonomous Database option such as **Oracle Autonomous Database 1** in the Create Connection dialog. • **Off** – Hide the Oracle Autonomous Database option such as **Oracle Autonomous Database 1** in the Create Connection dialog. Default: Off Restart Required: No
Oracle BI Publisher	This feature enables connecting to an Oracle Analytics Publisher instance to create augmentations using data from that instance. See Load Data from Oracle Analytics Publisher into NetSuite Analytics Warehouse (Preview). • **On** – Display the **Oracle BI Publisher** option in the Create Connection dialog. • **Off** – Hide the **Oracle BI Publisher** option in the Create Connection dialog. Default: Off Restart Required: No
Oracle E-Business Suite On-Prem	This feature enables configuring a remote agent, loading data from on-premises Oracle eBusiness Suite (EBS), and building a complete managed data pipeline with that data. See Load Data from On-premises E-Business Suite into NetSuite Analytics Warehouse (Preview). • **On** – Display the **Oracle E-Business Suite On-Prem** option in the Create Connection dialog. • **Off** – Hide the **Oracle E-Business Suite On-Prem** option in the Create Connection dialog. Default: Off Restart Required: No

Feature	Description
Oracle Eloqua	This feature enables connecting to an Oracle Eloqua instance to bring business-relevant marketing and sales campaign data to build a managed data pipeline with data from that instance. See Connect with Your Oracle Eloqua Data Source (Preview). • **On** – Display the **Oracle Eloqua** option in the Create Connection dialog. • **Off** – Hide the **Oracle Eloqua** option in the Create Connection dialog. Default: Off Restart Required: No
Oracle Enterprise Data Management Cloud	This feature enables connecting to Oracle Enterprise Data Management Cloud service instance to build a managed data pipeline with data from that instance. See Load Data from Enterprise Data Management Cloud into NetSuite Analytics Warehouse (Preview). • **On** – Display the **Oracle Enterprise Data Management Cloud** option in the Create Connection dialog. • **Off** – Hide the **Oracle Enterprise Data Management Cloud** option in the Create Connection dialog. Default: Off Restart Required: No
Oracle EPM - Enterprise Data Management	This feature enables connecting to an Oracle EPM instance and building a complete managed data pipeline with that data. See Load Data from Enterprise Performance Management into NetSuite Analytics Warehouse (Preview). • **On** – Display the **Oracle EPM - Enterprise Data Management** option in the Create Connection dialog. • **Off** – Hide the **Oracle EPM - Enterprise Data Management** option in the Create Connection dialog. Default: Off Restart Required: No
Oracle EPM - Financial Close and Consolidation	This feature enables connecting to Oracle EPM instance and building a complete managed data pipeline with that data. See Load Data from Enterprise Performance Management into NetSuite Analytics Warehouse (Preview). • **On** – Display the **Oracle EPM - Financial Close and Consolidation** option in the Create Connection dialog. • **Off** – Hide the **Oracle EPM - Financial Close and Consolidation** option in the Create Connection dialog. Default: Off Restart Required: No
Oracle EPM - Planning and Budgeting	This feature enables connecting to Oracle EPM instance and building a complete managed data pipeline with that data. See Load Data from Enterprise Performance Management into NetSuite Analytics Warehouse (Preview). • **On** – Display the **Oracle EPM - Planning and Budgeting** option in the Create Connection dialog. • **Off** – Hide the **Oracle EPM - Planning and Budgeting** option in the Create Connection dialog. Default: Off Restart Required: No

Feature	Description
Oracle EPM - Profitability and Cost Management	This feature enables connecting to Oracle EPM instance and building a complete managed data pipeline with that data. See Load Data from Enterprise Performance Management into NetSuite Analytics Warehouse (Preview). • **On** – Display the **Oracle EPM - Profitability and Cost Management** option in the Create Connection dialog. • **Off** – Hide the **Oracle EPM - Profitability and Cost Management** option in the Create Connection dialog. Default: Off Restart Required: No
Oracle JD Edwards On-Prem	This feature enables connecting to the various data sources in the on-premises JD Edwards instance and building a complete managed data pipeline with that data. See Load Data from On-premises JD Edwards into NetSuite Analytics Warehouse (Preview). • **On** – Display the **Oracle JD Edwards On-Prem** option in the Create Connection dialog. • **Off** – Hide the **Oracle JD Edwards On-Prem** option in the Create Connection dialog. Default: Off Restart Required: No
Oracle JDBC	This feature enables the remote agent-managed connectivity to an Oracle database using JDBC. You can create upto five connections and they are denoted as **Oracle JDBC 1**, **Oracle JDBC 2**, **Oracle JDBC 3**, **Oracle JDBC 4**, and **Oracle JDBC 5**. See Load Data from Oracle Database Using JDBC into NetSuite Analytics Warehouse (Preview). • **On** – Display the Oracle JDBC option such as **Oracle JDBC 1** in the Create Connection dialog. • **Off** – Hide the Oracle JDBC option such as **Oracle JDBC 1** in the Create Connection dialog. Default: Off Restart Required: No
MongoDB	This feature enables connecting to a MongoDB instance to build a managed data pipeline with data from that instance. See Load Data from Mongo Database into NetSuite Analytics Warehouse (Preview). • **On** – Display the **MongoDB** option in the Create Connection dialog. • **Off** – Hide the **MongoDB** option in the Create Connection dialog. Default: Off Restart Required: No
MySQL Cloud	This feature enables connecting to a MySQL Cloud instance to build data augmentation with data from that instance. See Load Data from MySQL Cloud Database into NetSuite Analytics Warehouse (Preview). • **On** – Display the **MySQL Cloud** option in the Create Connection dialog. • **Off** – Hide the **MySQL Cloud** option in the Create Connection dialog. Default: Off Restart Required: No

Feature	Description
MySQL On-Prem	This feature enables connecting to a MySQL on-premises instance to build data augmentation with data from that instance. See Load Data from On-premises MySQL Database into NetSuite Analytics Warehouse (Preview). • **On** – Display the **MySQL On-Prem** option in the Create Connection dialog. • **Off** – Hide the **MySQL On-Prem** option in the Create Connection dialog. Default: Off Restart Required: No
Oracle PeopleSoft On-Prem - Campus Solutions	This feature enables connecting to the various data sources in the on-premises PeopleSoft instance and building a complete managed data pipeline with that data. See Load Data from On-premises PeopleSoft into NetSuite Analytics Warehouse (Preview). • **On** – Display the **Oracle PeopleSoft On-Prem - Campus Solutions** option in the Create Connection dialog. • **Off** – Hide the **Oracle PeopleSoft On-Prem - Campus Solutions** option in the Create Connection dialog. Default: Off Restart Required: No
Oracle PeopleSoft On-Prem - Financials	This feature enables connecting to the various data sources in the on-premises PeopleSoft instance and building a complete managed data pipeline with that data. See Load Data from On-premises PeopleSoft into NetSuite Analytics Warehouse (Preview). • **On** – Display the **Oracle PeopleSoft On-Prem - Financials** option in the Create Connection dialog. • **Off** – Hide the **Oracle PeopleSoft On-Prem - Financials** option in the Create Connection dialog. Default: Off Restart Required: No
Oracle PeopleSoft On-Prem - Human Resources	This feature enables connecting to the various data sources in the on-premises PeopleSoft instance and building a complete managed data pipeline with that data. See Load Data from On-premises PeopleSoft into NetSuite Analytics Warehouse (Preview). • **On** – Display the **Oracle PeopleSoft On-Prem - Human Resources** option in the Create Connection dialog. • **Off** – Hide the **Oracle PeopleSoft On-Prem - Human Resources** option in the Create Connection dialog. Default: Off Restart Required: No

Feature	Description
Oracle PeopleSoft On-Prem - Learning Management	This feature enables connecting to the various data sources in the on-premises PeopleSoft instance and building a complete managed data pipeline with that data. See Load Data from On-premises PeopleSoft into NetSuite Analytics Warehouse (Preview). • **On** – Display the **Oracle PeopleSoft On-Prem - Learning Management** option in the Create Connection dialog. • **Off** – Hide the **Oracle PeopleSoft On-Prem - Learning Management** option in the Create Connection dialog. Default: Off Restart Required: No
Oracle Transportation Management	This feature enables connecting to an Oracle Transportation Management Cloud Service instance and creating augmentations using that data. See Load Data from Oracle Transportation Management Cloud Service into Oracle NetSuite Analytics Warehouse (Preview). • **On** – Display the **Oracle Transportation Management** option in the Create Connection dialog. • **Off** – Hide the **Oracle Transportation Management** option in the Create Connection dialog. Default: Off Restart Required: No
QuickBooks Online	This feature enables connecting to a QuickBooks Online instance to create augmentations using data from that instance. See Load Data from QuickBooks Online into Oracle NetSuite Analytics Warehouse (Preview). • **On** – Display the **QuickBooks Online** option in the Create Connection dialog. • **Off** – Hide the **QuickBooks Online** option in the Create Connection dialog. Default: Off Restart Required: No
Remote Agent	Remote Agent application helps bring data from the on-premises applications like E-Business Suite and JD Edwards into Oracle NetSuite Analytics Warehouse. See Set up the Remote Agent to Load Data into NetSuite Analytics Warehouse (Preview). • **On** – Display the Remote Agent option in the Create Connection dialog. • **Off** – Hide the Remote Agent option in the Create Connection dialog. Default: Off Restart Required: No
SFTP	This feature enables connecting to a Secure FTP source and building a complete managed data pipeline with that data. See Load Data from a Secure FTP Source into NetSuite Analytics Warehouse (Preview). • **On** – Display the SFTP option in the Create Connection dialog. • **Off** – Hide the SFTP option in the Create Connection dialog. Default: Off Restart Required: No

Feature	Description
Shopify	This feature enables connecting to a Shopify instance and building a complete managed data pipeline with that data. See Load Data from Shopify into NetSuite Analytics Warehouse (Preview). • **On** – Display the Shopify option in the Create Connection dialog. • **Off** – Hide the Shopify option in the Create Connection dialog. Default: Off Restart Required: No
Snowflake	This feature enables connecting to a Snowflake instance and creating augmentations using that data. See Load Data from Snowflake into Oracle NetSuite Analytics Warehouse (Preview). • **On** – Display the **Snowflake** option in the Create Connection dialog. • **Off** – Hide the **Snowflake** option in the Create Connection dialog. Default: Off Restart Required: No
SQL Server	This feature enables connecting to the SQL Server instance to create augmentations using data from that instance. See Load Data from SQL Server into Oracle NetSuite Analytics Warehouse (Preview). • **On** – Display the **SQL Server** option in the Create Connection dialog. • **Off** – Hide the **SQL Server** option in the Create Connection dialog. Default: Off Restart Required: No
Taleo	This feature enables connecting to a Taleo instance and creating augmentations using that data. See Load Data from Taleo into NetSuite Analytics Warehouse (Preview). • **On** – Display the **Taleo** option in the Create Connection dialog. • **Off** – Hide the **Taleo** option in the Create Connection dialog. Default: Off Restart Required: No
Oracle Warehouse Management Cloud	This feature enables connection to Oracle Warehouse Management System (WMS) to bring business relevant WMS data and build data augmentations for Logistics use cases. See Load Data from Oracle Warehouse Management Cloud into Oracle NetSuite Analytics Warehouse (Preview). • **On** – Display the **Oracle Warehouse Management Cloud** option in the Create Connection dialog. • **Off** – Hide the **Oracle Warehouse Management Cloud** option in the Create Connection dialog.

Data Connectors Available By Default

Feature	Description
Google Analytics	This feature enables connecting to a Google Analytics instance and building a complete managed data pipeline with that data. See Load Data from Google Analytics into NetSuite Analytics Warehouse.

Feature	Description
Oracle Object Storage Service	This managed pipeline connects to Oracle Object Storage to ingest datasets into Oracle NetSuite Analytics Warehouse followed by data augmentation. See Load Data from Oracle Object Storage into NetSuite Analytics Warehouse.
Salesforce REST	This feature enables an administrator to connect to a Salesforce SaaS source and build a complete managed data pipeline with that data. See Load Data from Salesforce into NetSuite Analytics Warehouse.

Configure Advanced Options

You can set several advanced options using the Console.

Topics:

About Advanced Configuration

Administrators can set more advanced, service-level options through the Systems Settings page.

Topics:

Performance and Compatibility Options

You use these options to configure performance and compatibility settings between Oracle BI Enterprise Edition and Oracle Analytics.

System Setting	More Information
Query Limit Extension	Determines whether the query limit can extend to 60 minutes to accommodate the occasional, longer running query. • **On** — The query limit can be extended to 60 minutes. • **Off** — The **Maximum Query Limit** setting on this page is used and never extends. **Default:** Off **Edition:** Professional and Enterprise

System Setting	More Information
Mobile Data Watch Service Frequency	Specifies the frequency at which the Data Watch service must scan the server for changes based on the frequency of changes in your data sources. The default is 240 (4 hours). You can disable this service by changing this setting frequency to 0 or by toggling the **Mobile Watch Service Enabled** setting to off. **Valid Values:** 0-10139 **Default:** 240 **Edition:** Professional and Enterprise
Strong Datetime Type Checking	Specifies whether to enforce strict checking for date and time data types and whether to reject queries that contain incompatibilities in date and time data types. • **On** — Enforces strict checking for date and time data types. • **Off** — Relaxes strict checking for date and time data types. However, invalid queries or queries with severe date and time incompatibilities may still be still rejected. For example, date and time incompatibilities might be rejected if your relational database uses strict checking for those data types. **Default:** On **Edition:** Professional and Enterprise
Load Semantic Models Using Multiple Threads	Specifies if semantic models load using multiple threads. If you find that your large datasets load slowly and impact system processing times, enabling this option may improve performance. • **On** — Semantic models load in parallel. • **Off** — Semantic models don't load in parallel. **Default: Off** **Edition:** Enterprise only
Enable Auto Insights on Datasets	Specifies whether the Auto Insights feature is available when datasets are created or modified. • **On** — The Enable Insights option is available in the Dataset Inspect dialog and insights are automatically generated and available for workbooks that use datasets with the Enable Insights option selected. • **Off** — Auto Insights and its related features are disabled. **Default: On** **Apply Change Required:** No, but when you change this setting, it may take a few minutes to take effect. **Edition:** Professional and Enterprise
Enable Immediate Dashboard Rendering	Specifies whether to display available dashboard content immediately or wait until all the dashboard content is ready. • **On** — Display dashboard content immediately even if some content is unavailable. • **Off** — Wait for all the dashboard content to be ready before displaying content. **Default:** Off **Edition:** Enterprise only

System Setting	More Information
Mobile Data Watch Service Enabled	Specifies if the Data Watch service is active. • **On** — The Data Watch service is active and all mobile users can specify the threshold value for bring backs. • **Off** — The Data Watch service is inactive. **Default:** On **Edition:** Professional and Enterprise
Evaluate Support Level	Specifies who can issue database functions: `EVALUATE`, `EVALUATE_ANALYTIC`, `EVALUATE_AGGR`, and `EVALUATE_PREDICATE`. By default (`0`), the `EVALUATE` database functions are disabled. • **1** — Service administrators only. Users with the BI Service Administrator application role can invoke `EVALUATE` database functions. • **2** — Anyone. Any user who signs in to can invoke `EVALUATE` database functions. • **0** (or any other value) — No one. All `EVALUATE` database functions are disabled in . **Valid Values: 0**, **1**, **2** **Default: 0** **Edition:** Professional and Enterprise
Enable Database Analytics Node in Data Flows	Specifies whether the Database Analytics node is displayed in data flows. • **On** — The Database Analytics node is available in data flows so that data flow designers can apply database analytics functions to the data. • **Off** — The Database Analytics node isn't available in data flows. This prevents data flow designers from generating a potentially high number of SQL statements and slowing database performance. **Default:** On **Edition:** Professional and Enterprise
Restrict Data Export and Delivery	Restricts the maximum number of rows users can export or deliver by email in formatted and unformatted content. Data export and delivery limits depend on the size of your service. **Valid Values:** Maximum - no restriction, 90% of Maximum, 80% of Maximum, 70% of Maximum, 60% of Maximum, 50% of Maximum, 40% of Maximum, 30% of Maximum, 20% of Maximum, 10% of Maximum, Minimum - 1000 rows **Default: Maximum - no restriction** **Edition:** Professional and Enterprise
Maxiumum Query Limit (seconds)	Specifies the maximum length of time a single query can run before it's canceled and users see a timeout message. The default is 660 seconds (11 minutes). **Valid Values:** 60-660 **Default:** 660 **Edition:** Professional and Enterprise

System Setting	More Information
Brushing Enabled for Subject Areas	Specifies whether brushing is enabled by default for workbooks that use data from subject areas. • **On** — Brushing is on by default for workbooks that use subject area data. • **Off** — Brushing is off by default for workbooks that use subject area data. Users can override this setting in the workbook and canvas properties. **Default: On** **Edition:** Enterprise only
Cache Enable	Specifies whether data query caching is enabled or disabled. • **On** — Data caching is enabled. • **Off** — Caching is disabled. **Default**: On **Edition**: Professional and Enterprise
Brushing Enabled for Datasets	Specifies whether brushing is enabled by default for workbooks that use dataset data. • **On** — Brushing is on by default for workbooks that use dataset data. • **Off** — Brushing is off by default for workbooks that use dataset data. Users can override this setting in the workbook and canvas properties. **Default: On** **Edition:** Professional and Enterprise

Email Delivered by Agents Options

You can use these options to customize the way agents deliver email.

System Setting	More Information
Use RFC 2231 Encoding	Specifies how to encode MIME email parameters. By default, RFC 2047 is used. • **On** — Use RFC 2231 to encode MIME email parameter values. RFC 2231 supports multi-byte languages. Select On if you deliver emails that contain multi-byte characters and use an email server that supports RFC 2231, such as Microsoft Outlook for Office 365 or Google Gmail. • **Off** — Use RFC 2047 to encode MIME email parameter values. **Default: Off** **Edition:** Enterprise only
Use BCC	Specifies whether to include the names of email recipients in the To: or Bcc: line. By default, email recipients are added to the Bcc: line. • **On** — Add email recipients to the Bcc: line. Names of email recipients are hidden. • **Off** — Add email recipients to the To: line. Everyone who receives the email sees the recipient list. **Default: On** **Edition:** Enterprise only

System Setting	More Information
Safe Domains	If you want to restrict the email domain that Oracle Analytics can send emails to, enter the name of the domain. For example, `examplemaildomain.com`. Use a comma to separate multiple domain names. For example, `exampledomain1.com,exampledomain2.com`. By default, there are no restrictions. **Edition:** Enterprise only
Maximum Number of Recipients per Email	Specifies the maximum number of recipients allowed in the To: or Bcc: line for a single email. You can set the maximum number of email recipients to avoid some SMTP servers from filtering out these emails as spam. If the recipient list exceeds the set limit, the list is split into smaller lists with the maximum number of allowed recipients in each list. **Valid Values: 0-1024** **Default: 0** (unlimited number of email recipients) **Edition:** Enterprise only
Maximum Email Size (KB)	Specifies the maximum size (KB) of a single email. If you set a maximum email size, you can avoid situations when SMTP servers reject emails that are too large, and in the event that an email exceeds the set limit, the email recipients receive an error message instead of the agent failing and just alerting the email author. **Valid Values: 0-20480** **Default: 0** (unlimited email size) **Edition:** Enterprise only

Other Options

These system setting options in the Console enable you to set the behavior for a variety of actions such as database queries, default URLs, display defaults, and sorting.

System Setting	More Information
Hide Loading Messages	Specifies if a detailed message is displayed during data load processing. • **On** — Detailed loading messages are hidden and a simplified message **Loading...** is displayed instead. • **Off** — Detailed loading messages are displayed. **Default: Off** **Edition:** Professional and Enterprise
Portal Path	Specifies the path of the dashboard page that's displayed by default when users sign in to . For example, `/shared/<folder>/_portal/<name>`. You can specify a single path for all users and multiple paths by user role, for example `{"application role 1":"catalog dashboard path 1","application role 2":"catalog dashboard path 2","default":"catalog dashboard path 3"}`. This setting applies to all users, but users can override it after they've signed in. You can enter a maximum of 5,000 characters in this field. **Edition:** Enterprise only

System Setting	More Information
Enable Subrequest Shipping	Specifies if sub-requests to source databases are executed separately as standalone queries or executed together. By default, sub-requests are shipped separately which can improve performance if you execute complex reports with a large group of sub-requests, that is, you prefer to ship the sub-requests separately in multiple simplified queries rather than ship a large single complicated query all at once. In Oracle BI Enterprise Edition, the default is set to `NO`. If you used Oracle BI Enterprise Edition and want to retain the previous default behavior, set this property to `NO` to continue executing database sub-requests together. • **Default** — Database sub-requests are shipped separately. This is the same as the value **YES**. • **YES** — Database sub-requests are shipped separately. • **NO** — Database sub-requests are shipped together, all at once. **Default: Default** **Edition:** Professional and Enterprise
Sort Order Locale	Applies to content migrated from Oracle BI Enterprise Edition. After you migrate content from your Oracle BI Enterprise Edition environment to , you may experience different sorting behaviors in analyses. For example, if you look at a migrated analysis in Polish, the upper case and lower case letters might sort based on the default locale, not the original Oracle BI Enterprise Edition locale. To preserve the Oracle BI Enterprise Edition sort behavior in , change this setting to **Polish**. **Edition:** Professional and Enterprise
Locale	Applies to content migrated from Oracle BI Enterprise Edition. After you migrate content from your Oracle BI Enterprise Edition environment to , you may see a different language in messages, dates, or currencies within analyses. For example, if you look at a migrated analysis in Polish, the currencies or dates might display based on the default locale, not the original Oracle BI Enterprise Edition locale. To preserve the Oracle BI Enterprise Edition currencies and dates in , change this setting to **Polish**. **Edition:** Professional and Enterprise
Enforce Safe Domains in Actions	Determines whether action links that users add to analyses and dashboards can invoke any URL or only URLs that administrators specify in the safe domains list. • **On** — Don't allow actions to invoke any URL that's not in the safe domain list. • **Off** — Allow actions to invoke any URL, even if the URL isn't listed as a safe domain. **Default: On** for a brand new service and **Off** for an existing service. **Apply Change Required** : No **Edition:** Enterprise only

System Setting	More Information
Recursive Datetime Type Checking	Specifies whether to enforce strict recursive data type checking for comparisons between identical data types (for example, integer to integer) or non-compatible data types (for example, integer to short integer) on all data sources or with all datasets. • **On** — Enforces strict recursive checking for identical or non-compatible data types on all data sources or datasets. • **Off** — Relaxes strict recursive checking for date and time data types on all data sources or datasets. However, if there are too many data type inconsistencies, you may want to change the data types to be compatible or use constants of the correct data type when comparing a column to a value. For example, after you migrate content from Oracle BI Enterprise Edition to , you might start seeing this type of check error in your reports because eary versions of Oracle BI Enterprise Edition didn't enforce strict checks: `[nQSError: 22024] A comparison is being carried out between non-compatible types <type1> and <type2>.` **Default: On** **Edition:** Professional and Enterprise
Repeat Rows on Excel Reports for Tables and Pivots	Specifies whether cells that span rows and cells that span columns are repeated when exporting tables and pivot tables to Excel. • **On** — If switched on, cells that span rows and cells that span columns are repeated, regardless of the Value Suppression setting in the Analysis editor. • **Off** — If switched off, the Value Suppression setting in the Analysis editor is honored and cells that span rows and cells that span columns don't repeat when exporting tables and pivot tables to Excel. **Default: Off** **Edition:** Enterprise only
Sort Null Values First	Specifies whether to sort NULL values before other values (On) or after (Off). Select the value that matches your database. If this setting doesn't match your database setting, then the database setting takes precedence. • **On** — Sorts NULL values before other values. • **Off** — Sorts NULL values after other values. **Default: Off** **Edition:** Professional and Enterprise

System Setting	More Information
Disable Right Trim for VARCHAR Data	Specifies whether the automatic removal of trailing spaces from varchar columns is enabled (Off) or disabled (On). For example, when this property is enabled (Off), when a user starts entering values in a field, the filter dialog automatically trims any trailing spaces. • **On** — Preserves trailing whitespaces in varchar columns. If you primarily use Oracle Database sources, you might want to keep the default Oracle Database behavior of preserving trailing whitespaces rather than removing them. When you toggle this property on, you avoid the overhead of trimming spaces, and this can improve performance. If you disable this property (set it to **On**) and you construct a filter such as PRODUCT_DESCRIPTION = 'My Product '), you must make sure the amount of trailing whitespace used exactly matches the varchar column value. If you don't, the filter won't correctly match the data values. • **Off** — Trims trailing whitespaces in varchar columns when processing queries. This is the default for . For example, if a user enters the text 'My Product ', it trims it to 'My Product'. **Default: Off** **Edition:** Professional and Enterprise
Use Vanity URL to Share Content in Email	Specifies the Oracle Analytics Cloud URL format that's used to share links to workbook visualizations in scheduled emails. If your organization set up a vanity URL for your system, enter the existing vanity URL that you want to use in the format: `https://myvanity.com/ui/` Alternatively, leave the setting blank to use the standard URL format in emails. See Share Visualizations Using Workbook Email Schedules. **Edition:** Professional and Enterprise

Preview Options

Administrators can turn some preview features on and off. This way, your organization can evaluate and learn how to use new features before they roll out by default.

System Setting	More Information
Enable Geometry Type	Specifies whether the geometry data type is available for data columns. • **On** — Enables you to use geometry columns in map visualizations and spatial geometry calculations. • **Off** — Disables the geometry data type. **Default:** Off **Apply Change Required:** Yes **Edition:** Enterprise only

Security Options

Use Security options to control how users can perform specific actions in analyses and dashboards.

These options apply only to analyses and dashboards. They don't apply to data visualizations.

System Setting	More Information
Enable Push Notifications	Specifies if mobile application push notifications are enabled (on) or disabled (off). • **On** — Enables mobile application push notifications to receive alerts and messages. • **Off** — Disables mobile application push notifications to stop receiving alerts and messages. **Default:** Off **Edition:** Professional and Enterprise
Save Workbook Thumbnail	To help people identify workbook content, Oracle Analytics can display thumbnail images for workbooks on the Home page. The information that's shown in these thumbnails is blurred to protect sensitive data from being exposed to users that don't have the same access as data authors. This setting overrides any **Save thumbnails** value set in the Workbook Properties dialog at the individual workbook level. **Note:** This setting doesn't apply to watchlists because they don't use thumbnails. Instead, watchlists display miniaturized visualizations that reload whenever you refresh the Home page. • **On** — Display blurred workbook thumbnails on the Home page. If this setting enabled (on), workbook owners can hide the thumbnail for individual workbooks if they need to. • **Off** — Don't display any workbook thumbnails on the Home page. Instead, show the standard icon for all workbooks. **Default: On** **Edition:** Professional and Enterprise
User Inactivity Timeout (minutes)	Specifies the number of minutes users are inactive before their browser or mobile connection must be re-authenticated. • **Valid Values**: 5 - 480 • **Default**: 60 • **Edition**: Professional and Enterprise
Sign Out Inactive Users Automatically	Specifies whether to automatically sign out users after the inactivity timeout is reached. • **On** — Users are automatically signed out when the inactivity timeout is reached. • **Off** — Users remain signed in even if the inactivity timeout is reached. • **Default**: Off • **Edition**: Professional and Enterprise

System Setting	More Information
Export Data to CSV and Tab-Delimited Files as Text	Specifies if leading apostrophes are added when data is exported to CSV or tab-delimited files, so all fields are treated as text. • **On** — Leading apostrophes are automatically added to CSV and tab-delimited files during exports. • **Off** — Data is exported to CSV files as-is. **Caution:** When **Off**, opening exported CSV files may invoke unwanted formulas. **Note:** This setting applies only to visualizations and analyses. It doesn't apply to pixel-perfect reports. **Default:** Off **Edition:** Professional and Enterprise
URL for Browser Script Actions	Specifies the URL for the JavaScript file containing custom Browser Script Actions. **Edition:** Enterprise only

System Setting	More Information
Allow HTML/JavaScript/CSS Content	Determines whether users can apply and save HTML, JavaScript, and CSS markup in various text fields for analyses and dashboards, and how any previously saved markup is used. • **Always** — Enables users to apply markup. Displays the **Contains HTML/JavaScript/CSS Markup** option in dialogs where additional formatting might be useful. For example: – For analyses: Various dialogs in the analysis editor, Analysis Properties dialog, Column Properties (Column Format) dialog, Edit Column Formula dialog, Narrative dialog, Ticker dialog, Static Text dialog, and New Calculated Measure dialog. – For dashboards: Various dialogs in the dashboard editor, Text Properties dialog, and Edit Header and Edit Footer dialogs (under Print and Export Options). • **Never** — Prevents users from applying markup. Hides the **Contains HTML/JavaScript/CSS Markup** option. Users can only enter plain text. Oracle Analytics ignores any markup that users previously entered and saved for their analyses and dashboards. • **HTML Only** — Enables users to apply HTML markup. Displays the **Contains HTML/JavaScript/CSS Markup** option in dialogs where additional formatting might be useful but only safe HTML is allowed (no JavaScript or CSS). When an analysis or dashboard opens, Oracle Analytics sanitizes any markup that users have entered and applies only the HTML markup. • **On Open** — Prevents users from applying additional markup (existing markup is retained). Hides the **Contains HTML/JavaScript/CSS Markup** option so users can enter only plain text. Any previously saved markup for analyses and dashboards continues to be applied. **Note:** The **On Open** option was previously named "Off". **Edition:** Enterprise only

View Options

You can use these options to configure default search and viewing settings for users working with analyses and dashboards.

System Setting	More Information
Default Scrolling Enabled	Specifies how data scrolls in tables, pivots, heat matrix, and simple and advanced trellis visualizations. • **On** — Data displays with a fixed header and content scrolling controls for users to browse the data. • **Off** — Data displays with content paging controls for users to browse the data. **Default:** On **Edition:** Enterprise only

System Setting	More Information
View Interactions: Display/ Hide Running Sum	Specifies whether the **Display/Hide Running Sum** option is selected by default in the Analysis Properties dialog: Interactions tab. • **On** — The **Display/Hide Running Sum** option is selected by default in the Analysis Properties dialog: Interactions tab. • **Off** — The **Display/Hide Running Sum** option isn't selected by default in the Analysis Properties dialog: Interactions tab. **Default:** Off **Edition:** Enterprise only
View Interactions: Display/ Hide Sub-totals	Specifies whether the **Display/Hide Sub-totals** option is selected by default in the Analysis Properties dialog: Interactions tab. • **On** — The **Display/Hide Sub-totals** option is selected by default in the Analysis Properties dialog: Interactions tab. • **Off** — The **Display/Hide Sub-totals** option isn't selected by default in the Analysis Properties dialog: Interactions tab. **Default:** Off **Edition:** Enterprise only
View Interactions: Drill	Specifies whether the **Drill** (when not a primary interaction) option is selected by default in the Analysis Properties dialog: Interactions tab. • **On** — The **Drill** (when not a primary interaction) option is selected by default in the Analysis Properties dialog: Interactions tab. • **Off** — The **Drill** (when not a primary interaction) option isn't selected by default in the Analysis Properties dialog: Interactions tab. **Default:** Off **Edition:** Enterprise only
View Interactions: Include/ Exclude Columns	Specifies whether the **Include/Exclude Columns** option is selected by default in the Analysis Properties dialog: Interactions tab. • **On** — The **Include/Exclude Columns** option is selected by default in the Analysis Properties dialog: Interactions tab. • **Off** — The **Include/Exclude Columns** option isn't selected by default in the Analysis Properties dialog: Interactions tab. **Default:** On **Edition:** Enterprise only
Prompt Auto Complete Matching Level	Specifies whether the auto-complete functionality uses matching to find the prompt value that the user enters into the prompt field. This setting doesn’t apply if the user accesses the Search dialog to locate and specify a prompt value. • **StartsWith** — Searches for a match that begins with the text that the user types. For example, the user types `M` and the following stored values are displayed: MicroPod and MP3 Speakers System. • **WordStartsWith** — Searches for a match at the beginning of a word or group of words. For example, the user types `C` and the following values are displayed: ComCell, MPEG Camcorder, and 7 Megapixel Digital Camera. • **MatchAll** — Searches for any match within the word or words. **Valid Values:** StartsWith, WordStartsWith, MatchAll **Default:** MatchAll **Edition:** Enterprise only

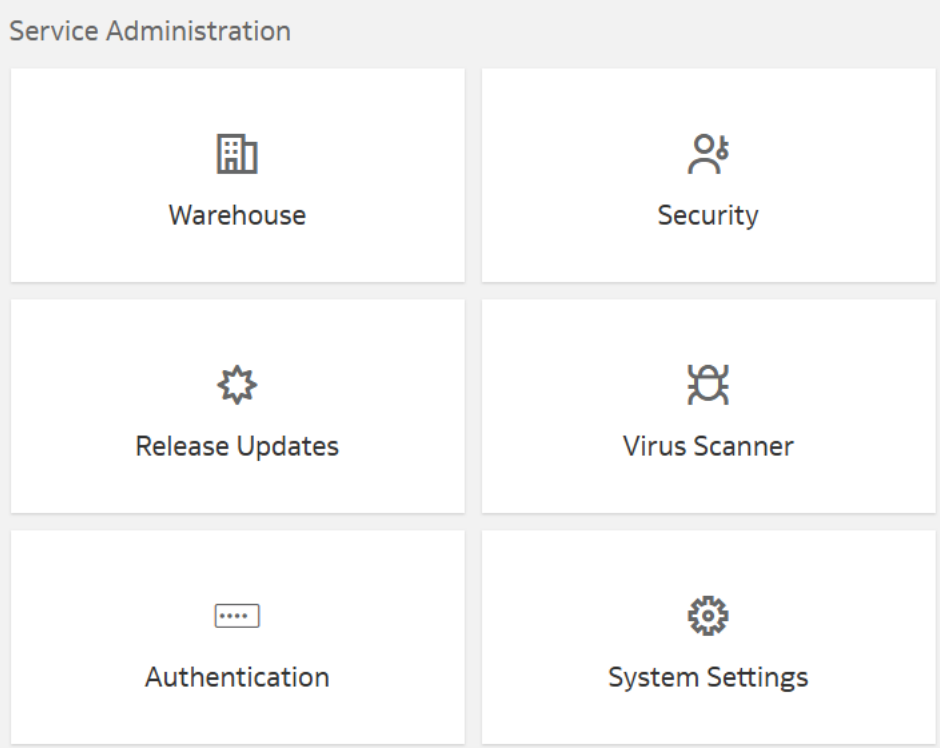
System Setting	More Information
View Interactions: Add/ Remove Values	Specifies whether the **Add/Remove Values** option is selected by default in the Analysis Properties dialog: Interactions tab. • **On** — The **Add/Remove Values** option is selected by default in the Analysis Properties dialog: Interactions tab. • **Off** — The **Add/Remove Values** option isn't selected by default in the Analysis Properties dialog: Interactions tab. **Default:** Off **Edition:** Enterprise only
Enable Enrichments in Workbooks	Specifies whether workbook editors can add dataset enrichments to a visualization directly from the Data Panel. This setting enables enrichments in workbooks for all users. Workbook editors who own a dataset or have editing privileges for it can enable or disable knowledge enrichments for that dataset using the **Enable Knowledge Enrichments** option. • **On** — Workbook editors can drag and drop enrichment based data elements to visualization canvases. • **Off** — Knowledge enrichments are not available for datasets. **Default:** On **Edition:** Professional and Enterprise
View Interactions: Create/Edit/ Remove Calculated Items	Specifies whether the **Create/Edit/Remove Calculated Items** option is selected by default in the Analysis Properties dialog: Interactions tab. • **On** — The **Create/Edit/Remove Calculated Items** option is selected by default in the Analysis Properties dialog: Interactions tab. • **Off** — The **Create/Edit/Remove Calculated Items** option isn't selected by default in the Analysis Properties dialog: Interactions tab. **Default:** Off **Edition:** Enterprise only
Table/Pivot View: Maximum Visible Rows	Specifies the maximum number of rows you want displayed for content paging in table and pivot table views in analyses and dashboards. The minimum number of rows you can specify to display is 100. **Valid Values:** 100-5000 **Default:** 5000 **Edition:** Enterprise only
View Interactions: Move Columns	Specifies whether the **Move Columns** option is selected by default in the Analysis Properties dialog: Interactions tab. • **On** — The **Move Columns** option is selected by default in the Analysis Properties dialog: Interactions tab. • **Off** — The **Move Columns** option isn't selected by default in the Analysis Properties dialog: Interactions tab. **Default:** On **Edition:** Enterprise only

System Setting	More Information
View Interactions: Sort Columns	Specifies whether the **Sort Columns** option is selected by default in the Analysis Properties dialog: Interactions tab. • **On** — The **Sort Columns** option is selected by default in the Analysis Properties dialog: Interactions tab. • **Off** — The **Sort Columns** option isn't selected by default in the Analysis Properties dialog: Interactions tab. **Default:** On **Edition:** Enterprise only
Enable Personalization in Workbooks	Specifies whether users can personalize workbooks. • **On** — Content designers can enable or disable personalization options (**Filter** and **Parameter**) in their workbooks. • **Off** — Personalization options aren't available for workbooks. **Default:** On **Edition:** Professional and Enterprise

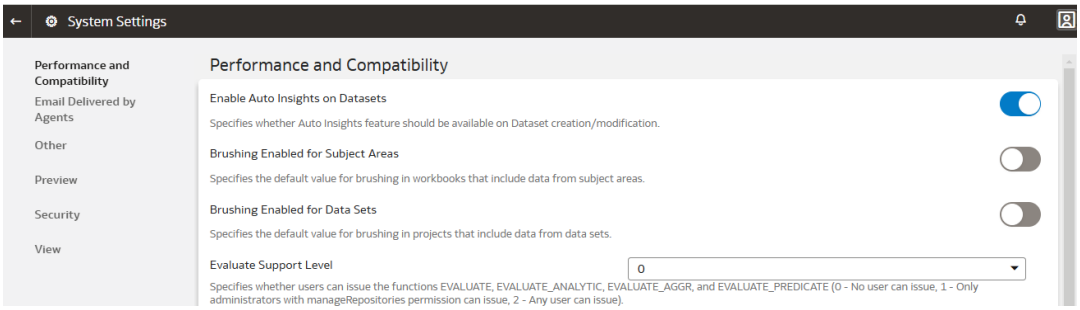
Set Advanced Options

Use the service instance console to set advanced options for Oracle NetSuite Analytics Warehouse.

1. In Oracle NetSuite Analytics Warehouse, click **Console**.
2. Under **Service Administration**, click **System Settings**.



3. On the System Settings page, update the toggle option for the applicable advanced option.



Create Additional NetSuite Connections

You can create additional connections to the NetSuite source based on your customer tier. These additional connections enable you to bring in data from multiple NetSuite accounts.

You may want to connect to additional NetSuite accounts if you've:

- One NetSuite primary account with live data and other NetSuite accounts having static data.
- More than one NetSuite account with live data but no data mash-up required.
- More than one NetSuite account with live data requiring data mash-up.

As a premium license tier Oracle NetSuite Analytics Warehouse customer, you can connect to two additional NetSuite accounts. If you're an enterprise license tier customer, then you can connect to ten additional NetSuite accounts. Based on your license tier, Oracle displays the additional NetSuite connections on the Create Connection dialog.

1. In NetSuite, complete these tasks:
   - Enable the Multi-Instance Connector (MIC) feature.
   - Add an integration record.
   - Create a token-based authentication permission and add it to a user.
   - Create a token.

   See Merging Your NetSuite Accounts Data Into Your NetSuite Analytics Warehouse Instance
2. In Oracle NetSuite Analytics Warehouse, **Console**, click **Data Configuration** under **Application Administration**.
3. On the Data Configuration page, click **Manage Connections**.
4. On the Manage Connections page, click **Create** and then click **Connection**.
5. In Create Connection, select **Data Extraction** in **Usage Type** and then select **NetSuite 2** or an applicable NetSuite option such as NetSuite 3 or NetSuite 4 as the connection type.
6. In **Create Connection**, enter these details and then click **Save**:
   - **Notification Email**: Enter an email address to receive notifications.
   - **User Name**: Enter your user name.
   - **JDBC URL**: Enter the following URL: jdbc:ns://<*NS Account ID*>.connect.api.NetSuite.com:1708;ServerDataSource=NetSuiteDW;Encrypted=1;CustomProperties=(AccountID=<*NS Account ID*>;RoleID=57;Uppercase=1). Replace <*NS Account ID*> with your NetSuite Account ID.

- **Account ID**: Enter your NetSuite Account ID.
- **Consumer Key**: Paste the Consumer Key / Client ID string you noted down after creating the integration record.
- **Consumer Secret**: Paste the Consumer Secret / Client Secret string you noted down after creating the integration record.
- **Token Key**: Paste the Token ID string you noted down after creating the token.
- **Token Secret**: Paste the Token Secret string you noted down after creating the token.

7. Verify that the **Refresh Metadata** toggle is enabled to ensure that the metadata is refreshed when you save the connection.

   You can later refresh the metadata from the Actions menu on the Manage Connections page, if required.

   > **Note:**
   >
   > You can't create augmentations for NetSuite 2 unless you perform a metadata extract.

8. On the Manage Connections page, select **Actions** for the NetSuite 2 connection and then select **Test Connection**.
9. After the connections are successfully established, navigate to the Data Configuration page, select the connection in **Data Source**, then click the Data Augmentation tile, and create a data augmentation using the NetSuite 2 data. Select the applicable NetSuite 2 source tables. See Augment Your Data.

Typical Workflow to Get Started with Oracle NetSuite Analytics Warehouse

To get started with Oracle NetSuite Analytics Warehouse for the first time, follow these tasks as a guide.

Task	Description	More Information
Enable NetSuite Analytics Warehouse	As an administrator, enable NetSuite Analytics Warehouse in the NetSuite account of the users.	Enabling NetSuite Analytics Warehouse, *NetSuite Analytics Warehouse*
Add and delete users.	As a user, after your administrator enables NetSuite Analytics Warehouse in your NetSuite account, add and delete users. Adding users to your NetSuite Analytics Warehouse account enables the users to work with data transferred to the data warehouse. You can add existing NetSuite users and non-NetSuite users.	Adding Users to NSAW, *NetSuite Analytics Warehouse*

Task	Description	More Information
Change the user account for data transfer credentials.	By default, NetSuite Analytics Warehouse assigns the data transfer credentials to the user that enabled NetSuite Analytics Warehouse. Oracle recommends that you use a generic email address such as user_account_data_transfer@example.com to ensure that the transfer credentials remain valid if the user that enabled NetSuite Analytics Warehouse leaves the organization.	Resetting NetSuite Analytics Warehouse Transfer Credentials, *NetSuite Analytics Warehouse*
Set up the data transfer in the NetSuite user interface.	As a user, begin transferring data to the NetSuite Analytics Warehouse by selecting the functional area.	Setting Up the Data Transfer, *NetSuite Analytics Warehouse*
Alternatively, set the pipeline parameters and create a data pipeline for a functional area to transfer data to the warehouse using the Data Configuration page in the NetSuite Analytics Warehouse user interface.	You can transfer data to the warehouse using the Data Configuration page. You specify the pipeline and report parameters and then create a data pipeline for a functional area.	Set Up the Pipeline Parameters Set Up the Reporting Configurations Create a Data Pipeline for a Functional Area
Start analyzing your data	After the data is transferred to the autonomous data warehouse, you can analyze your data using the capabilities of Oracle Analytics Cloud that is included with NetSuite Analytics Warehouse.	How Do I Visualize Data? (OAC)

Additionally, as an advanced user, you can perform these tasks:

Task	Description	More Information
Augment your data	You can supplement the data in your reports by extending the transaction entity, creating custom dimensions and custom facts.	Augment Your Data
Integrate third-party data	You can integrate data from other sources with the NetSuite data. Through this integration, you can create KPIs, metrics, cards, and dashboards using data from multiple sources.	Integrate Third-Party Data

2

Configure Oracle NetSuite Analytics Warehouse Data

You can set up data transfer using the NetSuite user interface or alternatively use the Data Configuration tile on the Oracle NetSuite Analytics Warehouse Console.

To set up data transfer using the NetSuite user interface, see Setting Up the Data Transfer.

In Oracle NetSuite Analytics Warehouse, use the Data Configuration page to specify the data load and reporting configuration details, and create data pipelines for functional areas that determine how your NetSuite data is loaded and displayed in the data warehouse. To navigate to the Data Configuration page:

1. Sign in to your NetSuite account.
2. Click **Setup**, click **Integration**, and then click **Configuration**.
3. On the NetSuite Analytics Warehouse Configuration page, click **Log in to NetSuite Analytics Warehouse**. You see the Oracle NetSuite Analytics Warehouse home page.
4. On the home page, open the **Navigator** menu, click **Console**, and then click **Data Configuration** under **Data Administration**. You see the Data Configuration page on which you can perform all the tasks discussed in this section.

Topics:

- View Load Request History
- View the Audit Log
- About Augmenting Your Data
- Augment Your Data
- Create a Dimension Alias
- Connect to an Autonomous Data Warehouse
- Disable Data Pipeline
- About Managing Data Connections
- Schedule Frequent Refreshes of Data
- Schedule Periodic Full Reload of Functional Area Data
- Schedule Frequent Refreshes of Warehouse Tables (Preview)
- Override Data Pipeline Schedules for Functional Areas (Preview)
- Prioritize Datasets for Incremental Refresh (Preview)
- Reset and Reload the Data Source
- Extend Data with Custom Data Configurations

Typical Workflow to Configure Data

Follow these tasks as a guide to configure Oracle NetSuite Analytics Warehouse data.

Task	Description	More Information
Specify the pipeline parameters	Set up the pipeline parameters for your data model file before running your data pipelines for the functional areas.	Set Up the Pipeline Parameters
Create a data pipeline for a functional area	To start analyzing your data, create data pipelines for the functional areas to copy data to the data warehouse.	Create a Data Pipeline for a Functional Area
Activate a data pipeline for a functional area	You must activate the data pipeline for a functional area to run and load data into the data warehouse.	Activate a Data Pipeline for a Functional Area
Specify the reporting configurations	Set up the reporting configurations for your data model file after activating a data pipeline for a functional area.	Set Up the Reporting Configurations
View request history	View the load request history by functional area, load type, and status.	View Load Request History
Augment your data	Augment your reports by using datasets that you create with specific columns from various data stores.	Augment Your Data

About Transaction Entities

Oracle NetSuite Analytics Warehouse is organized into logical business groupings called functional areas, which contain tables specific to the transaction types in each functional area.

Table Types

Each type of table follows a naming pattern for all functional areas.

- <Transaction Type> are transaction type stripes. In the NetSuite universal module definition (UMD), the transaction table is striped by transaction types using the TYPE column. NetSuite Analytics Warehouse creates tables by stripes. Transaction type can be SalesOrd, CustInvc, and so on, but <Transaction Type> contains an expanded and readable format such as _SALES_ORDER_ or _CUSTOMER_INVOICE_ respectively.
- <NS Table Name> are NetSuite tables which store dimensional data in the source.

Table 2-1 NetSuite Analytics Warehouse Table Types

Table Type	Naming Pattern	Source
Fact — Transaction Header	DW_NS_<Transaction Type>_F	Derived from NetSuite source tables transaction
Fact — Transaction Lines	DW_NS_<Transaction Type>_LINES_F	Derived from NetSuite source tables transaction, transactionline, and transactionaccountingline
Fact — Snapshot	DW_NS_<Transaction Type>_SNAPSHOT_F	Derived from transaction, systemnote, transactionhistory, and other transactions which act as snapshot events for the transaction
Dimensions	DW_NS_<NS Table Name>_D	Usually derived directly from source table
General Helper Table	DW_NS_<NS Table Name>_G	Usually derived directly from source table

Fact Denormalization

Fact — Transaction Lines are usually denormalized and contain additional attributes and dimension keys from the Fact — Transaction Header table to aid ease of reporting. For example, the Sales Order Lines fact contains status and trandate, which are denormalized from the Sales Order Header fact.

Fact — Transaction Header also contains dimensions keys or attributes from the Fact — Transaction Lines mainline in the NetSuite source tables that are non-transactional in nature.

Dimensions

The dimensions keys used in a fact usually follow the rule where they are named similar to the dimension table with which they are supposed to join to. For example, `DW_NS_SALES_ORDER_F.CUSTOMER` joins to `DW_NS_CUSTOMER_D.ID`.

Dimensions can be common across functional areas and transaction types. For example, `customer` and `item` are some of the common dimensions in NetSuite Analytics Warehouse.

Dimension History

The dimensions used in NetSuite Analytics Warehouse are type 1. Any changes in the source are reflected in NetSuite Analytics Warehouse either incrementally (if they are high volume dimensions) or in full update mode.

Base and Accounting Book Amounts for Transaction Amounts

Usually, transaction currency-based amount columns extracted from a source are converted to a subsidiary or base currency in NetSuite Analytics Warehouse fact lines tables. For example, `foreignamount` has a corresponding `base_amount`, `estgrossprofit` has a corresponding `base_estgrossprofit`, and so on.

Similarly, in line facts, the accounting book currency-based amounts are also converted into base and transaction currency when applicable. For example, `accountingbook_credit` and `accountingbook_debit` have `base_creditamount` and `base_debitamount` (in base or subsidiary currency) along with credit and debit (in transaction currency). The lines facts also consist of `accountingbook_currency` along with the `base_currency` key as a role-playing dimension in the lines fact tables. This helps in identifying the currency used for these measure columns.

Header facts contain conversion for the base or subsidiary currency amounts only for corresponding transaction currency amounts, such as `foreigntotal`, which will have a corresponding `base_total` along with the `base_currency` key. Header facts contain the `base_currency` key as a role-playing dimension. Header facts don't contain any accounting book amounts.

Flags Attributes

Both facts and dimension tables in NetSuite Analytics Warehouse consist of flag attributes which are useful for reporting purposes. The typical values in these columns is either `T` or `F`, indicating true or false. Alternately, `Y` or `N` indicates yes or no.

Snapshot Facts

NetSuite Analytics Warehouse contains snapshot facts for many transaction types. These tables track daily status and amount changes for a transaction type. The granularity of the snapshots tables is a combination of transaction identifier and date. Snapshots usually contain one row for each day from the transaction start date to the transaction close date. If the transaction is open on the ETL run date, then snapshots are built until one day before the ETL run date.

Delete Transactions

NetSuite Analytics Warehouse supports deletion of transactions. When any transaction is deleted in the source system, the same is deleted in NetSuite Analytics Warehouse after an incremental refresh.

Warehouse Table Grain Identification

The granularity of each table (fact, dimension, etc.) can be determined by checking the `source_record_id` column. When filtered by a single value, the table should have only one record. As a convention, source record ID is placed immediately after the columns that constitute the grain of the table. For example, for `DW_NS_SALES_ORDER_LINES_F`, the granularity of the table consists of `transaction`, `id`, and `accountingbook`, hence the `source_record_id` is placed as the fourth column after the above three grain columns.

Source record source_record_id contains the concatenated string-converted value of the grain columns. Hence for the above example, DW_NS_SALES_ORDER_LINES_F.source_record_id = transaction~id~accountingbook (or 32451~2~1).

Extraction Dates and Load Volume Control

Both Fact – Transaction Header and Fact – Transaction Lines tend to be high-volume tables and may impact data extract performance. To control the extract data volume, users can set an initial extract date as part of pipeline parameters while performing the setup. This controls the extraction of data in the load. Incremental loads are controlled using the last modified date attribute in transaction and transactionlines tables. Many dimensions also extract data incrementally using the last modified date attribute, while some dimensions always extract in full extracts. The ones which load in full extracts are usually smaller dimensions,

Reporting Parameters

NetSuite Analytics Warehouse has one reporting parameter that needs to be set: subsidiary. The subsidiary parameter is used for consolidating transaction values from other subsidiaries' currencies to this reporting subsidiary currency.

About Data Pipelines for Functional Areas

Data pipelines for functional areas load data specific to a functional area into the data warehouse.

These pipelines hold configuration parameters specific to a functional area such as General Ledger functional area under Finance and Sales functional area under NetSuite.

Allowed Actions for Data Pipelines

Here's a list of the actions allowed for data pipelines when they are in various states. This information enables you to know which actions you can and can't perform when a data pipeline is in a particular state.

Pipeline status	Available actions	Additional information
Saved	Edit and Delete	You can activate a data pipeline using the Edit wizard or can delete the data pipeline.
InActive	Edit, Activate, and Delete	You can activate a data pipeline using the Edit wizard or can delete the data pipeline.
Activation Scheduled	Edit and Delete	If you edit a scheduled data pipeline, then the previous data pipeline is unscheduled and a new data pipeline is scheduled.
Activation in Progress	View	You can view the data pipeline in Read-only mode.

Pipeline status	Available actions	Additional information
Activation Complete	Edit, Delete, Reset, Deactivate, and Refresh Data	You can perform these actions: • Edit only the Attribute Selection, Column Options, Entity Options, Dimension Keys, Schedule, and Save steps. • After the Edit action is completed, the data pipeline is ready for Refresh Data or AdHoc run. View the information icon that's displayed. • Perform a reset that marks the data pipeline for data reset and view the displayed information icon. The next incremental job peforms the actual action or you can peform the Refresh Data action to immediately reset the data. • Refresh the data. If an incremental job isn't running, then the Refresh Data action initiates an AdHoc run immediately to refresh the data. It peforms a full refresh if you had edited the data pipeline or performed a reset earlier. • Deactivate the data pipeline to change its state to InActive.

About Data Refresh Performance

Oracle strives constantly to improve performance for data loading in pipelines.

The performance of loading data for your instance will vary. The time to complete data processing, both full warehouse loads and incremental data loads, depends on various factors. A data pipeline load includes the following:

- Extracting data from the Netsuite source.
- Loading the data into Oracle Autonomous Data Warehouse.
- Transforming the data into the prebuilt schema.

The time to complete each of these steps is determined by various factors including:

- The availability of the source system.
- The size and complexity of the source data.
- The activated functional areas.
- Custom SQL queries that impact Oracle Autonomous Data Warehouse.
- Your queries running concurrently.

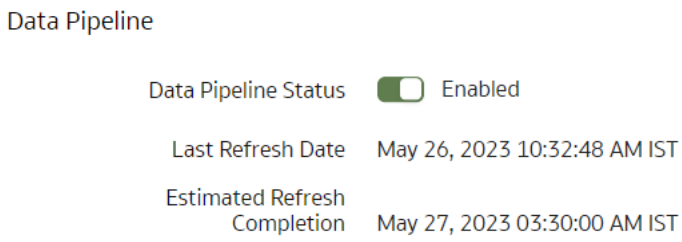
- Customizations made on the source system objects (which require a full load for those objects).
- The patching of source and target systems.

For daily refresh of data and pipeline execution to update all the resulting key metrics and dashboards, the execution time depends on many factors such as the amount of data getting incrementally refreshed. In order to plan your downstream workflow needs, ensure to perform these actions:

- Turn on the pipeline execution estimation using the Data Refresh Estimate preview feature on the Enable Features page. The estimated refresh completion details are visible on the Pipeline Settings page.
- Trigger any subsequent operations based on the estimated refresh completion time.
- Reach out to Oracle Support if the refresh isn't completed by the estimated completion time.

Data refresh is typically completed daily unless the data pipeline jobs are turned off or stuck. You can observe data loading times for your specific source and warehouse configuration to estimate the time it takes for an incremental daily refresh and for a full warehouse load. This information can help you plan for the optimal time in the day to start your daily data refresh. You may want to schedule the data load to run during off-peak hours, for example, run initial full warehouse loads during weekends and incremental loads during weeknights to ensure that users aren't impacted.

You can view the estimated refresh completion time for daily pipelines on the Pipeline Settings page in the Estimated Refresh Completion field as a Preview feature. This enables you to plan your tasks in the application.



The incremental data refresh process refreshes objects as base and derived datasets. The base datasets are objects that are part of the core pipeline and sourced directly from the Netsuite source and needed for core content (for example, transactions and dimensions). The derived datasets are based on data that require additional processing (for example, datasets used in prebuilt machine learning use cases). The incremental refresh process refreshes the base datasets first to ensure essential data is made available quickly. The system predicts and displays the incremental load completion time but doesn't do the same for derived datasets. If you haven't activated content that has derived datasets, then you won't see this information on the Pipeline Settings page in the Data Pipeline section.

About Pipeline Parameters

The pipeline parameters apply to all functional areas.

The pipeline parameters that apply at the functional area levels are initial extraction date and time to schedule the incremental job to run..

- Initial Extract Date: Initial extract date is used when you extract data for a full load. Transactional data created after the initial extract date processes and loads to the warehouse. It reduces the initial data load volume. After extracting the data for a functional area, avoid changing the initial extract date. If you need to change the initial extract data, then after changing the date, reset the data warehouse and reactivate the functional areas. See Reset the Data Warehouse.
- Data Refresh Schedule: Specify the frequency and when you want the incremental data load to happen. While specifying the timezone, the recommendation is to use city names to handle the daylight savings. For example, instead of selecting timezone such as EST or PST, select Europe/Bucharest or America/Los_Angeles. In this case, the data refresh process calculates the value mentioned in the Time field based on the local time irrespective of daylight savings.

Set Up the Pipeline Parameters

Set up the pipeline parameters for your data model file before running your data pipelines for the functional areas.

1. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration**.
2. On the Data Configuration page, under **Configurations**, click **Pipeline Settings**.
3. Under Global Parameters, indicate the initial extract date from which to load the transaction data.
4. Under Data Refresh Schedule, select the interval and depending on the selected interval, specify the time, day, and month when you want the incremental data load to happen.
5. Click **Save**.

About Reporting Configurations

Set up the reporting configurations to specify how data is presented on the key metrics, workbooks, visualizations, dashboards, and reports in Oracle NetSuite Analytics Warehouse.

You must activate a data pipeline for a functional area to be able to set up the reporting configurations. Based on the functional area for which you've activated a data pipeline, you see the applicable reporting configurations. For example, if you've activated a data pipeline for a functional area in Oracle Fusion ERP Analytics, then you see the reporting configurations applicable for Oracle Fusion ERP Analytics.

About the Reporting Configuration for NetSuite Analytics Warehouse

You specify values for these parameters as you configure reporting for Oracle NetSuite Analytics Warehouse:

- Subsidiary: This parameter controls the subsidiary or subsidiaries used for reporting. If you select parent, all the subsidiaries are covered in the reporting layer.

Set Up the Reporting Configurations

Specify the report parameters for your data model file.

1. In Oracle NetSuite Analytics Warehouse **Console**, click **Reporting Configuration**.
2. On the Reporting Configuration page, under Global Parameters, in **Subsidiary**, select a subsidiary to transfer data from that subsidiary.

3. Click **Save**.

Create a Data Pipeline for a Functional Area

As a functional administrator, create a data pipeline for a functional area to copy data to the data warehouse. Use these steps to select and schedule runs to update the Oracle Autonomous Data Warehouse instance with the latest data.

Perform this task only after you have set up the pipeline and report parameters. You can schedule an activation of a functional area even if activation of another functional area is in progress. However, you can't update the activation of a functional area that's in progress.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
3. On the Data Configuration page, click your service. For example, under Applications, click **NetSuite Warehouse**.
4. On the service page, for example, the Data Configuration: Oracle NetSuite Analytics page, click **Create**.
5. On the NetSuite Warehouse page, use the wizard to select your application areas to transfer data to the warehouse, and then click **Next**.
6. Review the parameters and click one of the options:
   - **Cancel**: To cancel the data pipeline for the functional area.
   - **Save**: To save the data pipeline for the functional area but not activate it.
   - **Activate**: To schedule when to run the data pipeline for the functional area. See Activate a Data Pipeline for a Functional Area.

Oracle NetSuite Analytics Warehouse extracts data only from the initial extract date that you specify. For example, if the initial extract date is 10th March and you've added data on 9th March in your source, then the data added on 9th won’t be part of this data pipeline activation plan. You can see the data pipeline activation plan listed on the Data Configuration page.

Edit a Data Pipeline for a Functional Area

You can change any details of your data pipeline for a functional area prior to activating it.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
3. On the Data Configuration page, click your service. For example, under Applications, click **NetSuite Warehouse**.
4. On the service page, for example, the Data Configuration: Oracle NetSuite Analytics page, click the **Action** menu for the data pipeline for a functional area that you want to edit, and click **Edit**.
5. Make the changes, review the changes, and then click **Save** or **Activate**.

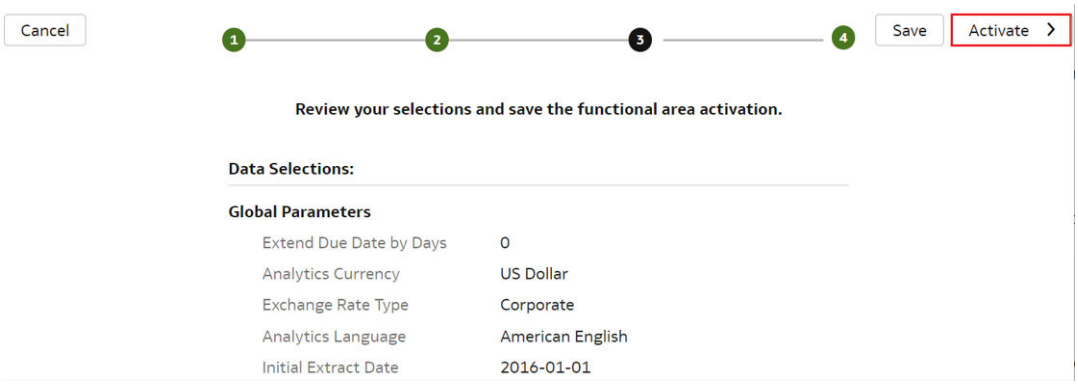
Activate a Data Pipeline for a Functional Area

You must activate the data pipeline for a functional area to run it and load the data into the data warehouse.

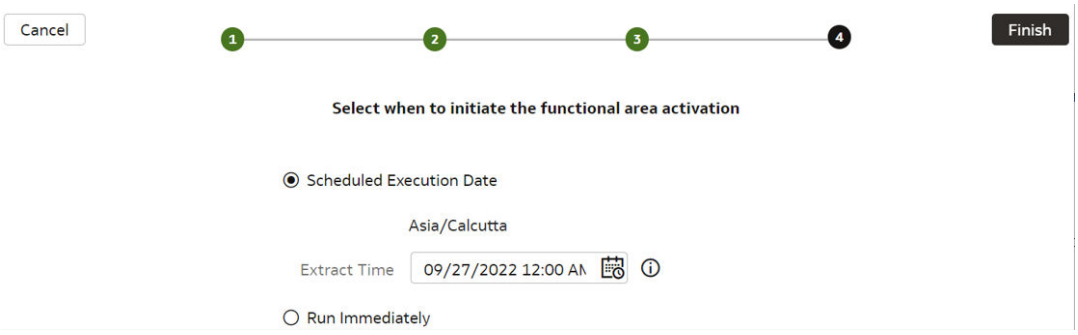
Ensure that you don't activate a data pipeline for a functional area in the following situations:

- Load in progress: If an incremental load is in progress.
- An impending load: If an incremental load is scheduled to run in the next hour.
- Exceeded the number of daily refresh requests: The maximum number of ad hoc data refresh requests for the day is four. If you've exceeded this number, then you can submit a request the following day.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
3. On the Data Configuration page, click your service. For example, under Applications, click **NetSuite Warehouse**.
4. On the service page, for example, the Data Configuration: Oracle NetSuite Analytics page, click the **Action** menu for the saved data pipeline for the functional area that you want to activate, and click **Edit**.
5. Review the details of the data pipeline for the functional area and then click **Activate**.



6. In step 4 of the Data Configuration wizard, select **Scheduled Execution Data** to specify the date and time on which to run the data pipeline for the functional area. Select **Run Immediately** to create and run the data pipeline for the functional area immediately. Click **Finish**.



Oracle NetSuite Analytics Warehouse runs the data pipeline for the functional area, loads data into your data warehouse, and displays your data pipeline for the functional area on the Data Configuration page. Once data is successfully loaded, the system updates the status of the data pipeline for the functional area to **Activation Completed**.

Activate Value-added Datasets

Oracle NetSuite Analytics Warehouse provides ready-to-use value-added datasets that load data independently into Oracle NetSuite Analytics Warehouse and aren't dependent on activation of any other functional areas. You must specifically activate these datasets.

The value-added datasets use data from multiple functional areas. For example, the Inventory Snapshot dataset sources data from multiple transaction types belonging to functional areas such as Sales, Purchases, and Inventory.

1. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration**.
2. On the Data Configuration page, under **Applications**, on the **NetSuite Warehouse** tile, click the drop-down and select a category such as **Oracle NetSuite Analytics for Finance and Supply Chain**.
3. On the Activations page, click **Create**.
4. Select a value-added dataset such as **Inventory Snapshot** and then click **Next**.
5. Review the parameters and click one of the options:
   - **Cancel**: To cancel the data pipeline for the functional area.
   - **Save**: To save the data pipeline for the functional area but not activate it.
   - **Activate**: To display the page for scheduling when to initiate activation of the functional area. Select **Scheduled Execution Date** to specify the date and time on which to run the data pipeline for the functional area. Select **Run Immediately** to create and run the data pipeline for the functional area immediately. After selecting your option, click **Finish**.

> **Note:**
>
> You can schedule an activation of a functional area even if activation of another functional area is in progress. However, you can't update the activation of a functional area that's in progress.

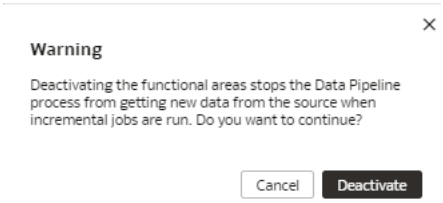
Deactivate a Data Pipeline for a Functional Area

You can deactivate all the incremental setup and jobs of a data pipeline for a functional area.

Deactivating a data pipeline for a functional area ensures that future incremental jobs don't select the specific functional area when the data pipeline runs. You can view the deactivated status of the data pipeline on the Data Configuration page.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.

3. On the Data Configuration page, click your service. For example, under Applications, click **NetSuite Warehouse**.
4. On the service page, for example, the Data Configuration: Oracle NetSuite Analytics page, click the **Action** menu for the active data pipeline for the functional area that you want to deactivate, and click **Deactivate**.
5. Review the warning message and then click **Deactivate**.

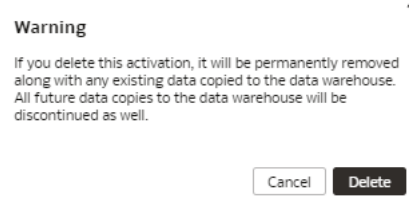


Delete a Data Pipeline for a Functional Area

You can delete a data pipeline for a functional area prior to activating it.

Deleting a data pipeline for a functional area permanently removes it, so data refreshes of the warehouse won't occur.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
3. On the Data Configuration page, click your service. For example, under Applications, click **NetSuite Warehouse**.
4. On the service page, for example, the Data Configuration: Oracle NetSuite Analytics page, click the **Action** menu for the data pipeline for the functional area that you want to delete, and click **Delete**.
5. Review the warning message and then click **Delete**.



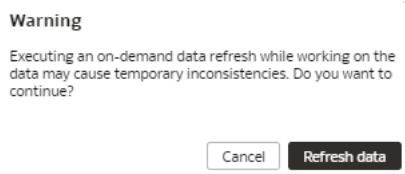
Refresh a Data Pipeline for a Functional Area

Refresh an activated data pipeline for a functional area to bring in new data for the selected functional area.

When you refresh the data pipeline, the system executes an incremental load immediately. The next scheduled run could take additional time to complete because of any new data. If you had reset a data pipeline for the functional area before refreshing data, then the system runs a full load immediately for that functional area.

> **Note:**
>
> Executing an on-demand data refresh while working on the data could cause temporary inconsistencies.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
3. On the Data Configuration page, click your service. For example, under Applications, click **NetSuite Warehouse**.
4. On the service page, for example, the Data Configuration: Oracle NetSuite Analytics page, click the **Action** menu for the data pipeline for the functional area whose data you want to refresh, and then click **Refresh Data**.
5. Review the warning message and then click **Refresh Data**.



Reload Data for a Data Pipeline

When you reload data for a data pipeline, you start the process of completely refreshing the data in the data warehouse for the functional area.

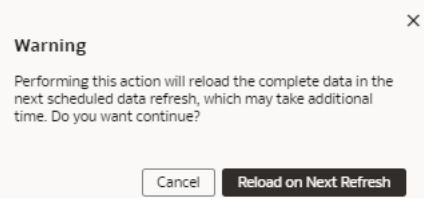
> **Note:**
>
> If you're reloading data for a functional area that depends on other functional areas, then you must also reload the data for the dependent functional areas to avoid incremental refresh failures. Alternatively, you can reset and reload the data source rather than reloading data for individual functional areas. See Reset and Reload the Data Source.

When you issue a request to reload data, the Request History page shows that the request is received and accepted. After you reload the data, the previous data still exists in the data warehouse related to the functional area. On the next data refresh, Oracle NetSuite Analytics Warehouse discards the existing data and loads new data. After reloading the data for the data pipeline, you see the data pipeline for the functional area on the Data Configuration page with the **Completed** status.

In a 24-hour period, you can reload the data for only 4 functional areas's data pipelines.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.

3. On the Data Configuration page, click your service. For example, under Applications, click **NetSuite Warehouse**.
4. On the service page, for example, the Data Configuration: Oracle NetSuite Analytics page, click the **Action** menu for the data pipeline for the functional area that you want to reload, and click **Reload Data**.
5. Review the warning message and then click **Reload on Next Refresh**.



Reset the Data Warehouse

Reset your data warehouse when your data source instance is refreshed from another environment, for example, when the environment is refreshed from production to a test environment. You may also want to reset the data warehouse when the data becomes corrupt.

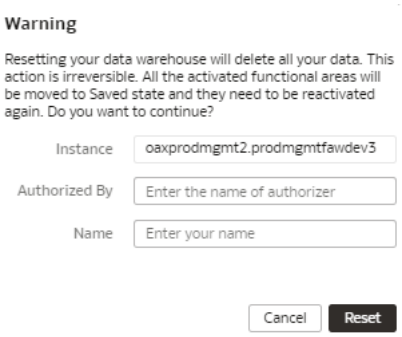
Prior to resetting the data warehouse, you must have at least one activated functional area. After resetting the data warehouse, the last refresh date shows the last increment details. You must reactivate all the data pipelines for the functional areas to load data. See Activate a Data Pipeline for a Functional Area

.

> **Note:**
>
> Your data isn't available until Oracle NetSuite Analytics Warehouse completes all the activations after resetting the data warehouse.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse Console, click **Enable Features**, click Generally Available tab, and then under Pipeline Features, enable **Reset Data Warehouse**.
3. Navigate to **Data Configuration** under **Application Administration** in the **Console**.
4. On the Data Configuration page, under **Configurations**, click **Pipeline Settings**.
5. On the Pipeline Settings page, click **Actions** and then click **Reset Data Warehouse**.

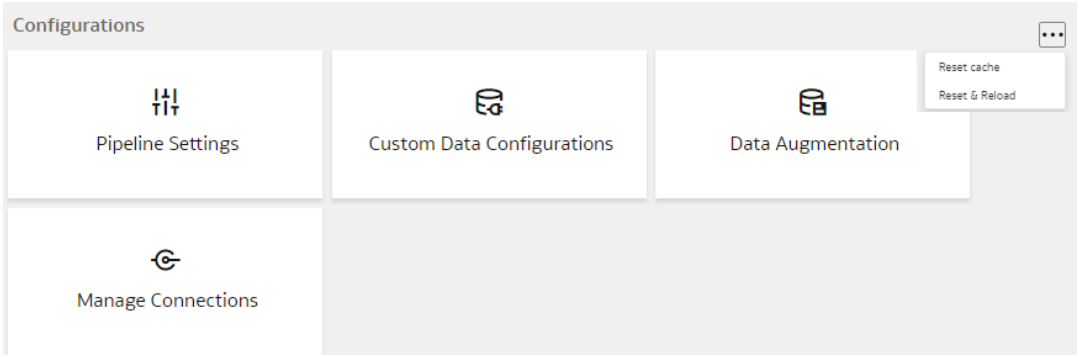
6. Review the warning message, provide the details, and then click **Reset**.



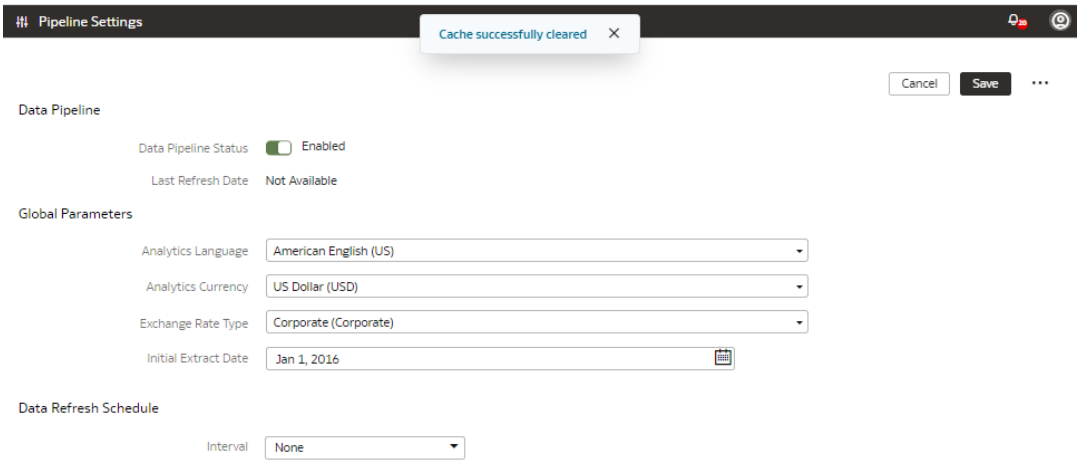
Reset the Cache

The data augmentation source column information is cached in the browser cache and is maintained for a week. If your source has new columns and they don't display in the Data Augmentation wizard, then you can reset the browser cache and retry data augmentation.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
3. On the Data Configuration page, click the **Reset Cache** menu option.



You see a message that cache has been cleared successfully.

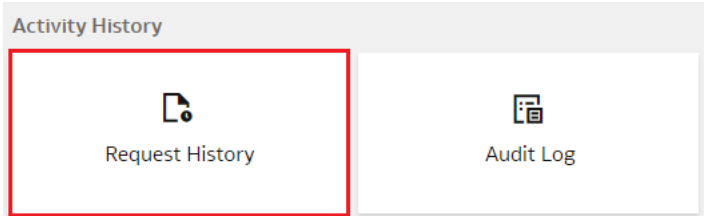


View Load Request History

You can view the data pipeline load request history by functional area, load type, and status along with other details for tracking and reference purposes.

The request history doesn't display the incremental loads. Because the request history shows only past load requests, any unscheduled loads don’t affect the existing pipelines and you can reactivate any functional area or augmentation.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
3. On the Data Configuration page, under **Activity History**, click **Request History**.



4. On the Request History page, view the history of loads that have been performed.

About Request Types and Statuses

When you perform tasks on a functional area, you can review the different types of requests involved with those tasks in the Request History area.

Request Types

Request Type	Description
Batch Request	An operation that performs multiple functions such as mapping attributes, or importing and export objects.
Content Patch	A content patch upgrade run.
Full Load (Adhoc)	A full load run that happens immediately upon request.
Full Load (Scheduled)	A request to schedule a full load run.
Map Attribute	A Netsuite-specific request run.
Module Reset	A request to delete an active functional area or source table.
Refresh Data	The system ran a request to refresh data.
Reset	A request to refresh the data in the data warehouse for the functional area.
Reset Data Warehouse	A request to reset the warehouse. This reset deletes all the customer-added warehouse data.
Target Model Upgrade	A request to upgrade the target model to the latest available version.

Request Statuses

Request Status	Description
Activation Completed	The job ran successfully and is now complete.
Activation in Progress	• The job is running. • The job is resolving.
Activation Scheduled	• Job is scheduled to run within one minute. • Job is scheduled to run at the specified date and time.
Deactivation Complete	The job is removed from Active status.
InActive	The job isn't saved or scheduled to run.
Received	The job request is submitted.
Saved	Job is saved but not scheduled to run.
Troubleshooting	The job is taking a long time and Oracle is investigating it.

View the Audit Log

You can view the data configuration-related actions that were performed such as activation of a data pipeline for a functional area, setting up of reporting parameters, and saving a data augmentation.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
3. On the Data Configuration page, under **Activity History**, click **Audit Log**.

4. On the Audit Log page, view the list of all the performed actions.

About Augmenting Your Data

Augment your reports by choosing specific columns from various custom transaction objects.

The data pipeline in Oracle NetSuite Analytics Warehouse extracts and loads data from NetSuite's standard transaction objects such as the standard records into the warehouse. Use the Data Augmentation option on the Data Configuration page in Oracle NetSuite Analytics Warehouse to extract and populate the following custom transaction types as a custom dimension, custom fact, or an extended entity into the warehouse:

- Custom attributes or records that reference a custom segment and custom list.
- Custom attributes that don't reference other objects.
- Custom reference that don't reference other objects.

Data augmentation allows you to seamlessly extract and load data from these custom transaction types and make it readily available in tables populated in Oracle Autonomous Data Warehouse. It also extends the subject areas in the semantic model and makes the custom transaction types available for reporting.

Augment Your Data

Extract and load data from your custom transactions and make it readily available in tables populated in the autonomous data warehouse.

You can use the system provided or customer provided source tables that are the custom transaction objects that you created in NetSuite. The system provided tables are pre-validated by Oracle NetSuite Analytics Warehouse. The customer provided tables are other source tables that are available for extraction but aren't validated by Oracle NetSuite Analytics Warehouse. As a user with the functional administrator or system administrator application role, you can allow usage of a particular table that isn't pre-validated by Oracle NetSuite Analytics Warehouse. However, Oracle can't ensure the success of processing such custom tables or any performance impacts, such as delays in the daily refreshing of data.

If you enable the **SME Options for Data Augmentation** under the Generally Available Features tab on the Enable Features page, then you can augment your reports with datasets created by extending an existing entity or group of facts, by adding a new dimension in the target instance, and by adding a new fact in the target instance. When you run these data augmentation pipeline jobs, they publish these datasets to the semantic model. However, this isn't the recommended practice. The recommended method is not to enable the **SME Options for Data Augmentation** feature and use the default **Dataset** augmentation type to bring varied data into the warehouse. When you run the Dataset data augmentation pipeline job, it doesn't publish anything to the semantic model. You can then use the semantic model extensions to create your own semantic model. This method supports complex semantic modelling to meet your business requirements. Use the Data augmentation capability to bring data into the warehouse and then use the Semantic Model Extensibility capability to create the joins and expose that data to the subject areas that you want. This enables flexibility and better performance of both the capabilities. Additionally, this method allows better lifecycle management. For example, if you need to make any adjustments to the semantic model, then you can make the changes directly in the semantic model. You don't need to adjust the data augmentation that brought the data into the warehouse.

The Dataset augmentation type isn't associated with any other augmentations. Based on the incremental schedule, the data in this dataset gets refreshed during scheduled pipeline refresh. But unlike other augmentations, this augmentation isn't linked to other augmentations, and you can't change the attributes as dimension or measure. This dataset isn't associated with any subject area, since it is simply copying the dataset from source and creating a warehouse table. You can perform semantic model extension after the table is created. To use this dataset to build the joins or incorporate an object from the dataset into your semantic model, you must run an incremental load prior to using it because the incremental load populates the dataset.

You see the data augmentation pipeline jobs on the Data Augmentation page with one of these statuses:

- Activation in Progress - You can't edit, delete, or schedule a data augmentation pipeline job while activation is in progress.
- Activation Completed - You can edit the data augmentation to add or delete VO attributes and save the changes. You can't modify the schedule in this status.
- Activation Scheduled - You can edit the data augmentation to add VO attributes, save the changes while retaining the existing schedule, reschedule the execution date and time, or execute the plan immediately.

> **Note:**
>
> You can change the names of the columns that you've added from the various data sources in your data augmentation. Later if you delete a data augmentation, then you must wait for the daily incremental run to complete to see the change in the reports, cards, and decks.

1. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
2. On the Data Configuration page, under **Configurations**, click **Data Augmentation**.
3. On the Data Augmentation page, click **Create**.

   You see the data augmentation wizard with multiple steps. You can create the Dataset augmentation type by default. However, if you enable the **SME Options for Data Augmentation** under the Generally Available Features tab on the Enable Features page, then you can select the **Create Dimension**, **Create Fact**, and **Extend Entity** type of augmentations. To create these augmentation types, ensure to enable the **SME Options for Data Augmentation** feature. See Enable Generally Available Features
4. In step 1 of the wizard, select an augmentation type. Each augmentation type requires you to complete certain tasks.
5. For the type of augmentation, if you select **Create Dimension** to add a new dimension in the target instance, follow these instructions:
   a. Select **Custom Record** in **Source Dataset Type**.
   b. Select **NetSuite Warehouse** in **Pillar**.
   c. In **Source Table Type**, specify the source table type using either of the options and then click **Next**:
      - Select **System Provided** and then in **Source Table**, select a table to which you want to add the new dimension.
      - Select **Customer Provided** and then in **Source Table**, enter the name of the table to which you want to add the new dimension.
   d. Optionally, select the **Versioned Dataset** check box to enable full load of the source table data everytime and then click **Next**.
   e. In step 2 of the wizard, in the Available Attributes pane, select the check box for the attributes that you want in your new dimension, and then click **Add Selected**.
   You see the attributes that you had selected in the Target attributes pane and the recommended (defined) primary key. You can either accept this key or override it with your own primary key definition. You can mark any columns as incremental to ensure that you get new data specific to the columns.
   f. Optionally, select **Advanced** to reorganize the order of columns that are marked as primary keys. To reorder, select a primary key row and then drag to define a suitable join order.
   g. Click **Next**.
   h. In step 6 of the wizard, provide the following details and click **Finish** to save and schedule your data augmentation pipeline job:
      i. Name your augmentation pipeline job; for example, `Sales Order`.

ii. Enter a suffix for the target table name using underscore in place of spaces between words and don't use special characters; for example, `Sales_Order_D`. The augmentation process automatically creates the target table name.

iii. Provide a description.

iv. Select the functional area and one or multiple subject areas in which you want to include this augmentation pipeline job.

v. Specify the options to save the data augmentation pipeline job without executing it, or schedule the execution date and time, or execute it immediately.

6. For the type of augmentation, if you select **Create Fact** to add a new fact table in the target instance, then follow these instructions:

   a. Select **Custom Record** in **Source Dataset Type**.

   b. Select **NetSuite Warehouse** in **Pillar**.

   c. In **Source Table Type**, specify the source table type using either of the options and then click **Next**:

      - Select **System Provided** and then in **Source Table**, select a table from where you want to add the new fact table.
      - Select **Customer Provided** and then in **Source Table**, enter the name of the table from where you want to add the new fact table.

   d. Optionally, select the **Versioned Dataset** check box to enable full load of the source table data everytime and then click **Next**.

   e. In step 2 of the wizard, in the Available Attributes pane, select the check box for the attributes that you want in your new fact table, and then click **Add Selected**.
   You see the attributes that you had selected in the Target attributes pane and the recommended (defined) primary key. You can either accept this key or override it with your own primary key definition. You can mark any columns as incremental to ensure that you get new data specific to the columns.

   f. Optionally, select **Advanced** to reorganize the order of columns that are marked as primary keys and specify one of your incremental keys to use when determining the initial extract date.

   g. Click **Next**.

   h. In step 6 of the wizard, provide the following details and click **Finish** to save and schedule your data augmentation pipeline job:

      i. Name your augmentation pipeline job; for example, `Sales Order Line`.

      ii. Enter a suffix for the target table name using underscore in place of spaces between words and don't use special characters; for example, `Sales_Order_F`. The augmentation process automatically creates the target table name.

      iii. Provide a description.

      iv. Select the functional area and one or multiple subject areas in which you want to include this augmentation pipeline job.

      v. Specify the options to save the data augmentation pipeline job without executing it, or schedule the execution date and time, or execute it immediately.

7. For the type of augmentation, if you select **Dataset** to copy a dataset as is into a target warehouse table, then follow these instructions:

   a. Select **Supplemental Data** in **Source Dataset Type**.

   b. In **Source Table Type**, specify the source table type using either of the options:

- Select **System Provided** and then in **Source Table**, select a table for the attributes.
- Select **Customer Provided** and then in **Source Table**, enter the name of the table for the attributes.

> **Note:**
>
> You can select your customer extensions, item extensions, vendor extensions, partner extensions, and employee extensions as source tables if you want to add attributes from these entity extensions.

c. In **Source Table**, select the applicable table whose attributes you want to add into the dataset and then click **Next**.

d. Optionally, select the **Versioned Dataset** check box to enable full load of the source table data everytime and then click **Next**.

e. In step 2 of the wizard, select the check box for the attributes from the source table to add to the target table, and then click **Next**.

f. In step 3 of the wizard, select the settings for the selected columns, and then click **Next**.

g. In step 6 of the wizard, provide the following details and click **Finish** to save and schedule your data augmentation pipeline job:

h. In step 3 of the wizard, map the extension attributes to the warehouse entities by selecting the warehouse entity that is being extended. You see the primary column of the warehouse entity mapped with the attribute that you are extending.

i. In step 6 of the wizard, provide the following details and click **Finish** to save and schedule your data augmentation pipeline job:

   i. Provide a name and description for your augmentation.

   ii. Enter a suffix for the target table name using underscore in place of spaces between words and don't use special characters; for example, `Customer_Deposit_EXT`. The augmentation process automatically creates the target table name.

   iii. Specify the options to save the data augmentation pipeline job without executing it, or schedule the execution date and time, or execute it immediately.

8. For the type of augmentation, if you select **Extend Entity** to add attributes to dimensions or measures to facts, then follow these instructions:

   a. Select **Custom Attribute** in **Source Dataset Type**.

   b. Select **NetSuite Warehouse** in **Pillar**.

   c. In **Source Table Type**, specify the source table type using either of the options and then click **Next**:

   - Select **System Provided** and then in **Source Table**, select a table for the attributes.
   - Select **Customer Provided** and then in **Source Table**, enter the name of the table for the attributes.

> **Note:**
>
> You can select your customer extensions, item extensions, vendor extensions, partner extensions, and employee extensions as source tables if you want to add attributes from these entity extensions.

d. Optionally, select the **Versioned Dataset** check box to enable full load of the source table data everytime and then click **Next**.

e. In step 2 of the wizard, in the Available Attributes pane, select the check box for the attributes from the source table that you want in your target table, and then click Add Selected. You see the attributes that you had selected in the Target attributes pane and the recommended (defined) primary key. You can either accept this key or override it with your own primary key definition.

f. Select Advanced to reorganize the order of columns that are marked as primary keys and specify a date or timestamp data type column as one of your incremental keys to use when determining the initial extract date.

g. Click **Next**.

h. In step 3 of the wizard, map the extension attributes to the warehouse entities by selecting the warehouse entity that is being extended. You see the primary column of the warehouse entity mapped with the attribute that you are extending.

i. In step 6 of the wizard, provide the following details and click **Finish** to save and schedule your data augmentation pipeline job:

   i. Name your augmentation pipeline job; for example, `Customer Deposit`.

   ii. Enter a suffix for the target table name using underscore in place of spaces between words and don't use special characters; for example, `Customer_Deposit_EXT`. The augmentation process automatically creates the target table name.

   iii. Provide a description.

   iv. Select the functional area and one or multiple subject areas in which you want to include this augmentation pipeline job.

   v. Specify the options to save the data augmentation pipeline job without executing it, or schedule the execution date and time, or execute it immediately.

Create a Dimension Alias

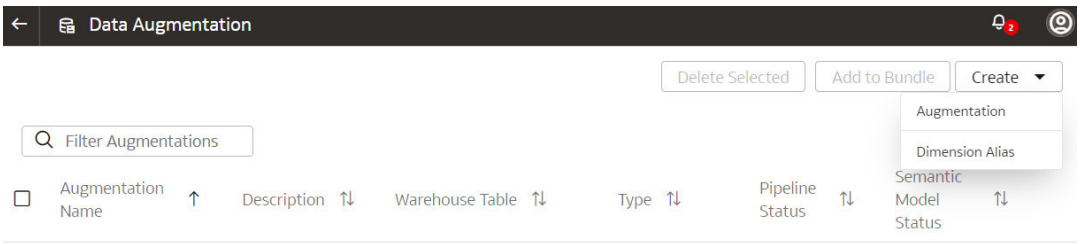
Dimension alias are alias names on the warehouse dimension tables. You can specify the alias names for the Dimension type augmentations and the data that's in the warehouse already as dimensions. The alias names enable you to reuse the existing warehouse tables with different names in the subject areas.

You create dimension aliases as a type of augmentation. This is useful when you want to reinstate a dimension for analysis. For example, if you would like to reuse the Vendor dimension as a Shipper dimension as well, you can create an alias for Vendor.

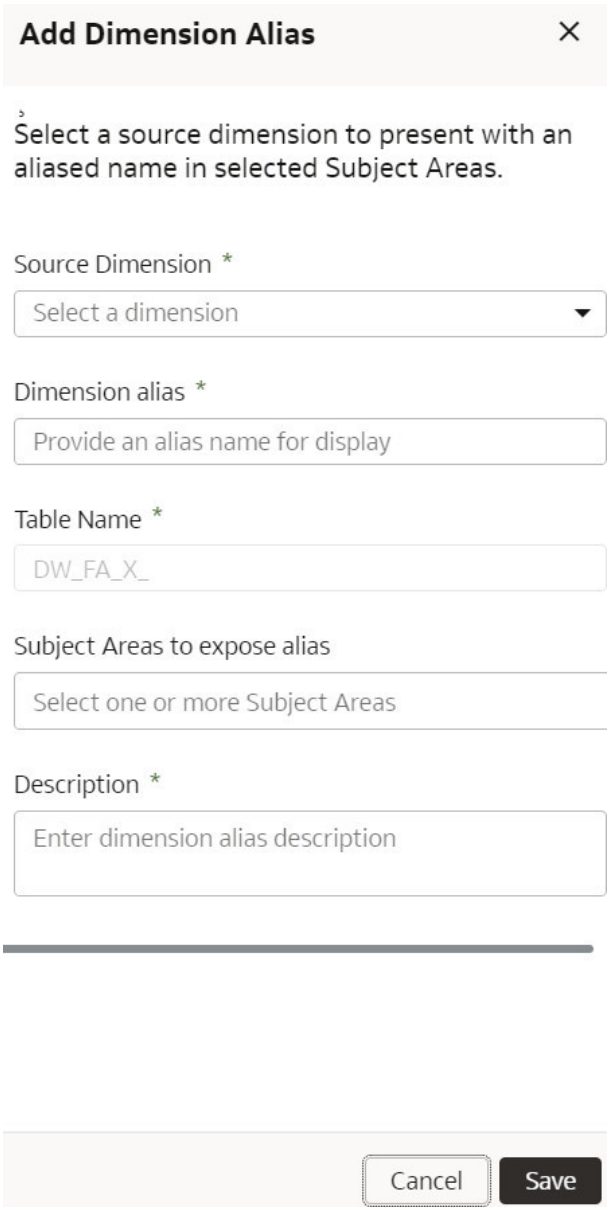
You can also select complex custom attributes as dimension aliases and retain the name defined in NetSuite. Ensure that you enable Dimension Alias in the Custom Attribute Mapper section under the Preview Features tab on the Enable Features page.

When you create the aliases, Oracle NetSuite Analytics Warehouse doesn't copy the data for the dimension aliases. These augmentations don't have a pipeline status, but they are visible in the subject area.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
3. On the Data Configuration page, under **Configurations**, click **Data Augmentation**.
4. On the Data Augmentation page, click **Create**, and select **Dimension Alias**.



5. In the Add Dimension Alias dialog, in **Source Dimension**, select a dimension from the source tables such as Employee.



6. In **Dimension Alias**, enter a name such as Shipper Dim.

   In **Table Name**, Oracle NetSuite Analytics Warehouse displays the name such as DW_FA_X_SHIPPER_DIM.

7. In **Subject Areas to expose alias**, select one or more subject areas from the subject areas where the original source dimension exists.
8. Add a description of the dimension alias.
9. Click **Save**.

Connect to an Autonomous Data Warehouse

NetSuite Analytics Warehouse provides three ways to connect to an Autonomous Data Warehouse: Oracle Analytics Cloud Connector, Database Actions (SQL Developer Web), and SQL Developer desktop client.

The Oracle Analytics Cloud connector provides a way to query database objects for analysis, visualization, and reporting. You can query database objects by configuring Oracle Analytics Cloud datasets or writing SQL manually.

Database Actions (SQL Developer Web) and SQL Developer desktop client allow you to create, edit, and query database objects. You can also use them to perform administrative tasks such as creating users and database schemas and resetting passwords. These tools help you investigate, debug, and validate data.

Oracle recommends using Database Actions (SQL Developer Web) to leverage the modern cloud interface and its additional features.

Topics:

Configure Autonomous Data Warehouse Credentials

You need to configure credentials of the OAX_USER and ADMIN users and download a wallet file to connect to an Autonomous Data Warehouse.

All connection methods initially require credentials for ADMIN and OAX_USER. You must reset these credentials annually, or configure them if you're performing this task for the first time.

The ADMIN can perform certain important tasks, such as changing OAX_USER's password, but can't see pipeline objects or data. The OAX_USER can browse and query pipeline objects and data. The following steps ensure that both users have correct credentials that you can use for connecting to an Autonomous Data Warehouse.

Topics:

About Autonomous Data Warehouse Wallet and Administrator Credentials

The Autonomous Data Warehouse Wallet contains certificate keys for accessing your Autonomous Data Warehouse instance.

The client credentials `.zip` that you download contains the following files:

- `cwallet.sso` - Oracle auto-login wallet.
- `ewallet.p12 - PKCS #12` wallet file associated with the auto-login wallet.
- `sqlnet.ora` - SQL*Net profile configuration file that includes the wallet location and TNSNAMES naming method.
- `tnsnames.ora` - The SQL*Net configuration file that contains network service names mapped to connect descriptors for the local naming method.

- Java Key Store (JKS) files - Key store files for use with JDBC Thin connections.
- README - Contains information about database tools and resources.

> **Note:**
>
> Wallet files, along with the database user ID and password, provide access to data in your Autonomous Data Warehouse. Store wallet files in a secure location. Share wallet files only with authorized users. If wallet files are transmitted in a way that might be accessed by unauthorized users (for example, over public email), transmit the wallet password separately and securely.

Reset the ADMIN User Password

Administrator credentials allow you to access and manage your Autonomous Data Warehouse. You can reset the ADMIN user's credentials, but if you reset the password, that might affect other users who log in as ADMIN.

1. Sign in to NetSuite Analytics Warehouse.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Warehouse**.
3. On the Warehouse page, under **Administrator Credentials**, click **Reset Credentials**, then provide and confirm the new password.

Download the Autonomous Data Warehouse Wallet

The Autonomous Data Warehouse Wallet contains certificate keys for accessing your Autonomous Data Warehouse instance.

1. Sign in to NetSuite Analytics Warehouse.
2. Click **Warehouse** on the **Console**.
3. On the Warehouse page, under **ADW Wallet**, click **Download Wallet**.
4. In the **Set Wallet Password** dialog, provide a password to encrypt the keys inside the wallet, then click **Download**.

   For more information about wallet contents, see About Autonomous Data Warehouse Wallet and Administrator Credentials.

> **Note:**
>
> Wallet files, along with the database user ID and password, provide access to data in your Autonomous Data Warehouse. Store wallet files and the wallet password in a secure location. Share wallet files only with authorized users. If wallet files are transmitted in a way that might be accessed by unauthorized users (for example, over public email), then transmit the wallet password separately and securely.

Get the Database Actions URL

You need to locate the URL for Database Actions to be able to reset the OAX_USER's password.

1. Unzip the downloaded wallet file, and open the README file in a text editor.
2. Find and copy the URL ending with `/sql-developer` listed in the **Database Actions** section of the README file.
3. Paste the URL in a browser, then sign in to the **Database Actions** web page.

Reset the OAX_USER Password

You need to reset the password for the OAX_USER before creating a connection to an Autonomous Data Warehouse.

1. Sign in to **Database Actions** with the ADMIN user's credentials.
2. Click **DATABASE USERS** in the **Administration** section and edit the OAX_USER account details.
3. Enter a new password and confirm it.
4. Click the **Web access** toggle, then click **Apply Changes**.
5. If Database Actions (SQL Developer Web) isn't accessible, use the SQL Developer desktop client to reset the OAX_USER password. Run the following commands, defining a new password in place of `NewPassword` in SQL Developer desktop client.

```
ALTER USER OAX_USER IDENTIFIED BY NewPassword;
COMMIT;
```

Connect to an Autonomous Data Warehouse with Database Actions (SQL Developer Web)

Follow these steps to connect to Autonomous Data Warehouse with Database Actions (SQL Developer Web).

Topics:

- Configure Autonomous Data Warehouse Credentials (if not already done)
- Query Tables for NetSuite Analytics Warehouse from Autonomous Data Warehouse as OAX_USER
- Query the Autonomous Data Warehouse System Tables as ADMIN

Query Tables for NetSuite Analytics Warehouse from Autonomous Data Warehouse as OAX_USER

Verify proper configuration of your Autonomous Data Warehouse connection by running a sample query.

1. Sign in to Database Actions as OAX_USER.
2. On the **Development** tab, click the **SQL** label to open SQL Developer Web. Select **All Objects** in the **Objects** dropdown.

   You should see all objects for OAX USER in the **Navigator**.

3. Run this query to verify proper configuration:

```
select * from DW_NS_ACCOUNT_D
```

Query the Autonomous Data Warehouse System Tables as ADMIN

Verify proper configuration of your Autonomous Data Warehouse connection by running a sample query.

1. Sign in to Database Actions as ADMIN.
2. On the **Development** tab, click the **SQL** label to open SQL Developer Web.
3. Run this query to verify proper configuration:

```
select sum(bytes)/1024/1024/1024/1024 as USED_STORAGE_TB from dba_segments
      where owner not in ('SSB', 'SH');
```

Connect to an Autonomous Data Warehouse with Oracle Analytics Cloud

Follow these steps to connect to an Autonomous Data Warehouse with Oracle Analytics Cloud.

Topics:

- Configure Autonomous Data Warehouse Credentials (if not already done)
- Configure the Connection for OAX_USER in Oracle Analytics Cloud
- Create an Oracle Analytics Cloud Dataset

Configure the Connection for OAX_USER in Oracle Analytics Cloud

You can connect to an Autonomous Data Warehouse with Oracle Analytics Cloud.

When you configure the connection, select the service name with the "low" prefix in its name. Connecting with the "high" or "medium" service might lead to performance issues in reports in NetSuite Analytics Warehouse and delays in completion of daily incremental data pipeline loads. See Database Service Names for Autonomous Transaction Processing and Autonomous JSON Database for more information.

1. Sign in to your NetSuite Analytics Warehouse instance.
2. Click **Go to Home Page**.
3. Click **Navigator** and then **Data**.
4. Click the **Connections** tab.
5. On the Oracle Analytics page, click **Create**, click **Connection**, then click **Oracle Autonomous Data Warehouse**.
6. Enter these values and then click **Save**:
   a. **Connection Name**: `ADW - OAX_USER`
   b. **Description**: `ADW - OAX_USER`
   c. **Client Credentials**: Click **Select**, then browse to select the wallet.zip file.
   d. **Username**: `OAX_USER` (default)
   e. **Password**: The OAX_USER password.

f. **Service Name**: Select **<service name>-low**. Using low prevents pipeline conflicts.

7. In the **Connections** tab, click the ellipsis icon to the right of the name of the connection you just created to open the **Actions** menu and click **Inspect**.
8. Click **Access** and then **Roles**.
9. Type `ServiceAdmin` in the **Search By Name** field, then click to add it.
10. Click the **Full Control** radio button for the ServiceAdmin user.
11. Click **Save** and then **Close**.

Create an Oracle Analytics Cloud Dataset

Verify proper configuration of your Oracle Analytics Cloud connection to the Autonomous Data Warehouse by creating a basic dataset.

See Create a Dataset From a Connection.

Connect to an Autonomous Data Warehouse with SQL Developer Desktop Client

You can optionally use the SQL Developer desktop client to connect to Autonomous Data Warehouse. You need to configure credentials, and then set up connections for both ADMIN and OAX_USER.

> **Note:**
>
> Oracle recommends using Database Actions (SQL Developer Web) to leverage the modern cloud interface and its additional features.

Topics:

- Configure Autonomous Data Warehouse Credentials (if not already done)
- Configure the Connection to Autonomous Data Warehouse for ADMIN with SQL Developer Desktop Client
- Configure the Connection to Autonomous Data Warehouse for OAX_USER with SQL Developer Desktop Client

Configure the Connection to Autonomous Data Warehouse for ADMIN with SQL Developer Desktop Client

You can use the SQL Developer desktop client to connect to Autonomous Data Warehouse. However, this connection method is optional, and using Database Actions (SQL Developer Web) is the preferred method.

When you configure the connection, select the service name with the "low" prefix in its name. Connecting with the "high" or "medium" service might lead to performance issues in reports in NetSuite Analytics Warehouse and delays in completion of daily incremental data pipeline loads. See Database Service Names for Autonomous Transaction Processing and Autonomous JSON Database for more information.

1. Launch SQL Developer.

2. In the **Connections** pane, click **New Connection**.
3. Create an SQL Developer connection for the ADMIN user by setting the following values in the **New/Select Database Connection** dialog:
   a. Define a **Name** for the connection, such as `NSAW_ADMIN`.
   b. Set the **Username** to `ADMIN`.
   c. Enter the password for the `ADMIN` user.
   d. Set the **Connection Type** to **Cloud Wallet**.
   e. For the **Configuration file**, click **Browse** and select the wallet.zip file.
   f. **Service**: Select **<service name>-low**. Using low prevents pipeline conflicts.
   g. When connecting with a virtual private network (VPN), enter details in the **Proxy** tab.
   h. Click **Test** to validate the connection.
   i. Click **Save**.
   j. Click **Connect**.
4. Run this query to verify proper configuration:

```
select sum(bytes)/1024/1024/1024/1024 as USED_STORAGE_TB from dba_segments
      where owner not in ('SSB', 'SH');
```

Configure the Connection to Autonomous Data Warehouse for OAX_USER with SQL Developer Desktop Client

You can use the SQL Developer desktop client to connect to Autonomous Data Warehouse.

When you configure the connection, select the service name with the "low" prefix in its name. Connecting with the "high" or "medium" service might lead to performance issues in reports in NetSuite Analytics Warehouse and delays in completion of daily incremental data pipeline loads. See Database Service Names for Autonomous Transaction Processing and Autonomous JSON Database for more information.

1. Launch SQL Developer.
2. In the **Connections** pane, click **New Connection**.
3. Create an SQL Developer connection for OAX_USER. Set the following values in the **New/Select Database Connection** dialog:
   a. Define a **Name** for the connection, such as `NSAW_OAX_USER`.
   b. Set the **Username** to `OAX_USER`.
   c. Enter the password for OAX_USER.
   d. Set the **Connection Type** to **Cloud Wallet**.
   e. For the **Configuration file**, click **Browse** and select the wallet.zip file.
   f. **Service Name**: Select **<service name>-low**. Using low prevents pipeline conflicts.
   g. When connecting with a virtual private network (VPN), enter details in the **Proxy** tab.
   h. Click **Test**.
   i. Click **Save**.
   j. Click **Connect**.

4. Expand **Synonyms** to see all of the prebuilt synonyms for OAX_USER.
5. Run this query to verify proper configuration:

```
select * from DW_NS_ACCOUNT_D
```

Disable Data Pipeline

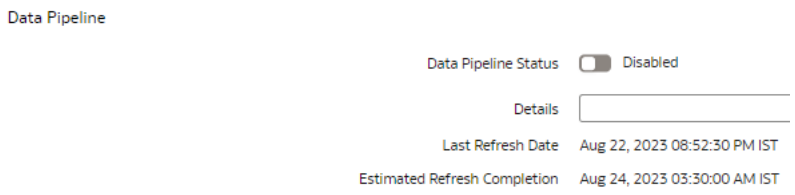
As the cloud account administrator with the functional administrator or system administrator application role, you can disable the data pipeline and enable it again.

You may want to disable the data pipeline in the following scenarios:

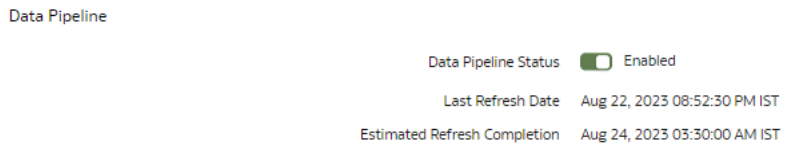
- If you don’t want to run pipeline jobs for a particular source.
- If you don’t want to run pipeline jobs for a particular duration such as a quite time in your business activities.

In cases where the pipeline jobs are failing due to issues, Oracle disables the data pipeline from the backend to investigate and resolve. On the Pipeline Parameters page, Oracle provides a reason and action for you, if the resolution needs to be done from your side. You can resolve the issue and as an administrator you can enable the data pipeline yourself using the **Data Pipeline Disabled** toggle.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
3. On the Data Configuration page, under **Configurations**, click **Pipeline Settings**.
4. On the Pipeline Settings page, under Data Pipeline, select the **Data Pipeline Status** toggle to **Disabled** and enter a reason in **Details**.



5. Select the **Data Pipeline Status** toggle again to enable the data pipeline after you've completed the reason for disabling the data pipeline.

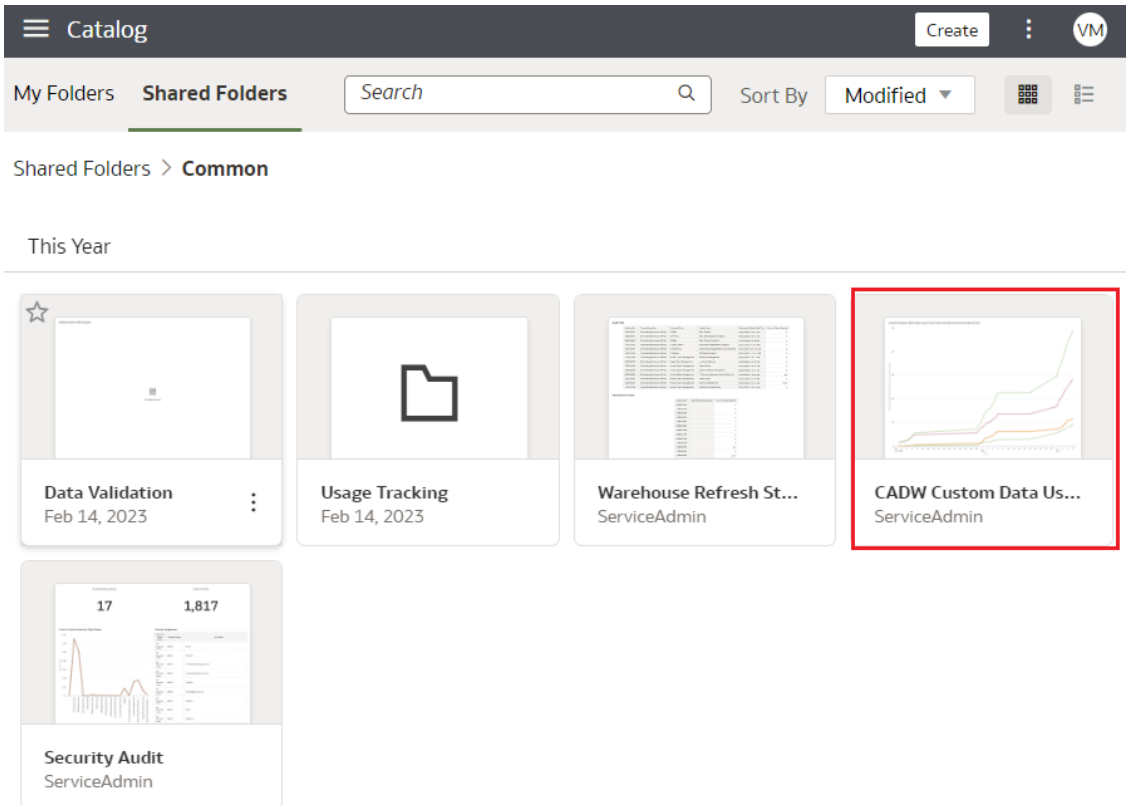


About Managing Data Connections

You can connect to a variety of data sources and remote applications to provide the background information for reports. You can blend the additional data from the various data sources with the prebuilt datasets to enhance business analysis.

Oracle NetSuite Analytics Warehouse can connect to other pre-validated data sources such as Oracle Object Storage, cloud applications such as Google Analytics, and on-premises applications such as Oracle E-Business Suite.

You can view the usage of capacity for custom data that's loaded into Oracle NetSuite Analytics Warehouse through the connectors in the Custom Data Usage dashboard available in the Common folder. The dashboard shows data loaded daily and monthly from each of the activated external data sources.



Topics

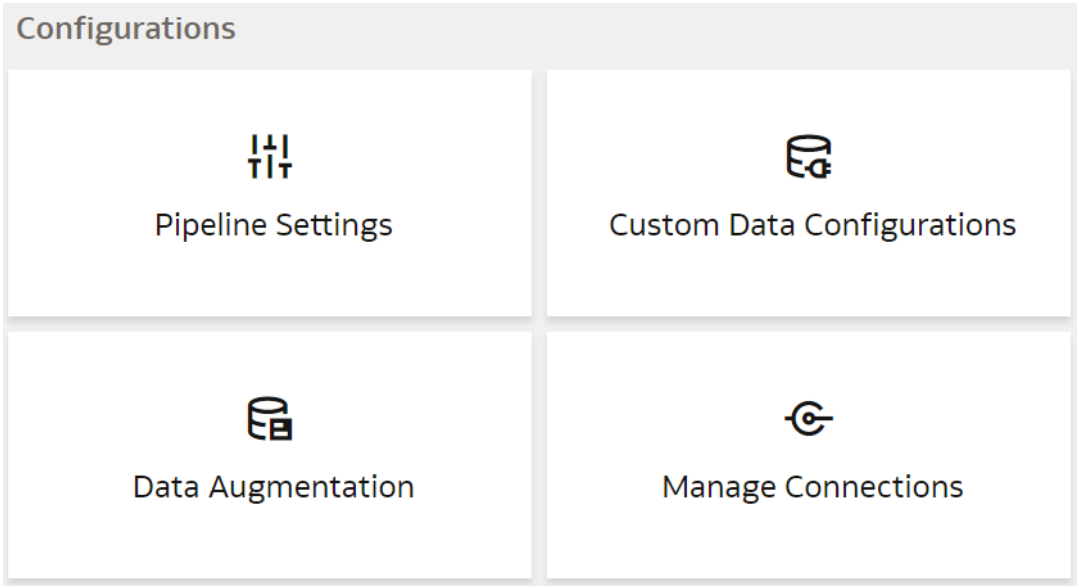
- Connect With On-premises Sources
- Connect with Cloud File Storage Sources
- Connect With Cloud Sources

Create a Data Connection Type

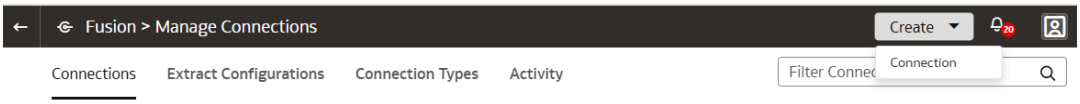
Connection Type specifies the source to which you're connecting. A connection type can have multiple connections.

You can create a custom data source type for any remote data connection.

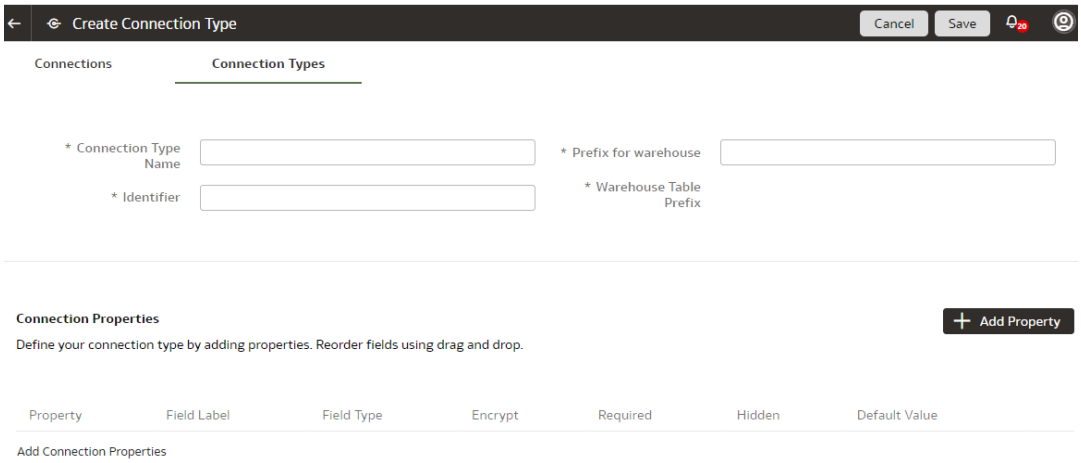
1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
3. On the Data Configuration page, click **Manage Connections** under **Configurations**.



4. On the Manage Connections page, click **Create** and then click **Connection Type**.



5. In the Create Connection Type dialog, enter the **Name**, **Identifier**, and **Prefix for warehouse** for the connection type.



6. Click **Add Property** and enter the parameters for each property that defines the connection.
7. When you've finished adding the connection properties, you can reorder them as needed.
8. Click **Save the Connection Type**.

The new connection is available on the Connections page.

Edit a Data Connection Type

If the properties or parameters for a data connection type change, you can edit them.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
3. On the Data Configuration page, click **Manage Connections** under **Configurations**.
4. On the Manage Connections page, click **Connection Types** and then click or search for the connection type you want to edit.

   You can't edit or delete Oracle-managed connections.
5. Click the **Action** button next to the connection type you want to change.
6. In the dialog box for the connection type, edit the details for your connection type, and then click **Save**.

Delete a Data Connection Type

You can delete a data connection type if you don't need it anymore.

> **Note:**
>
> After you delete a connection type, you can't create new data connections to it.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.

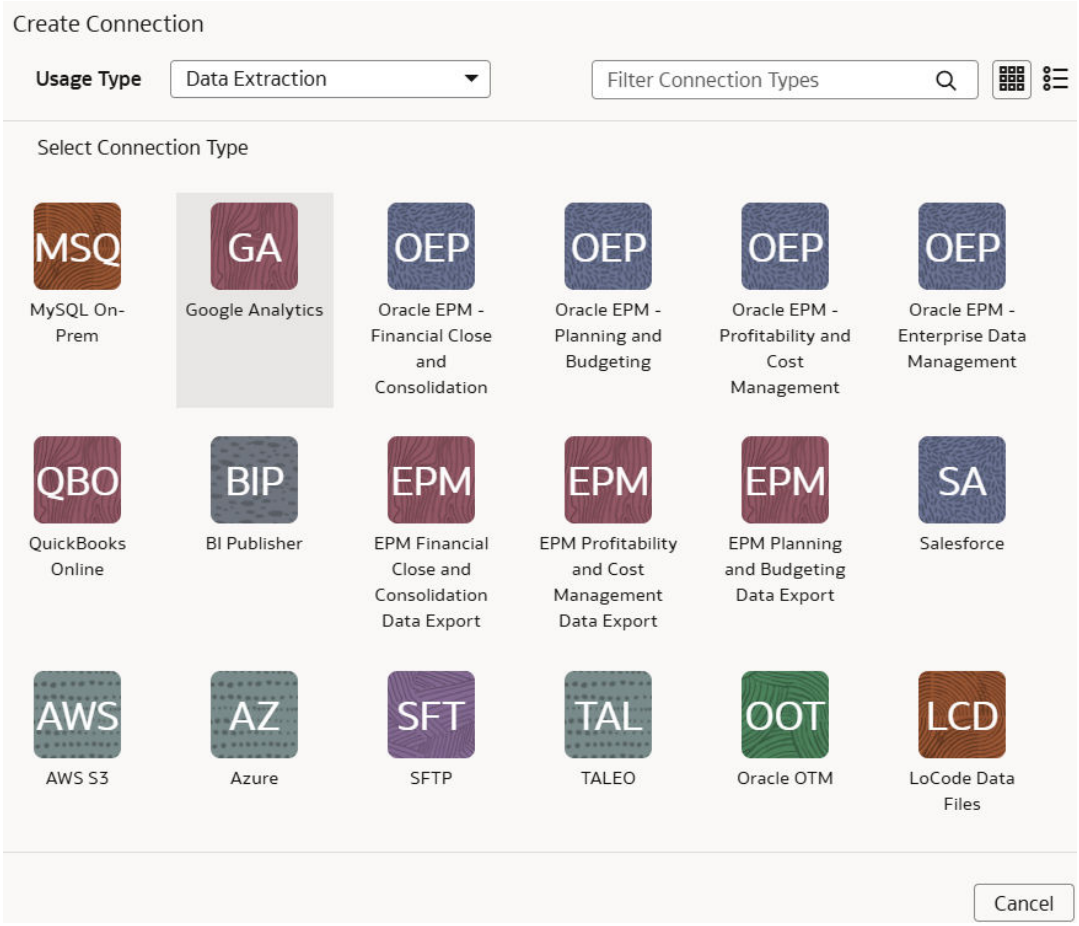
3. On the Data Configuration page, click **Manage Connections** under **Configurations**.
4. On the Manage Connections page, click **Connections** and then select or search for the connection you want to test.
5. Click the **Action** menu for the connection and select **Delete**.
6. In the Delete Connection dialog box, click **Delete**.

Create a Data Connection

You create a data connection to a data source to load data from that source into Oracle NetSuite Analytics Warehouse. You can create a connection for any available connection type.

While creating the connection, the system populates the connection name based on the connection source and you can't change it while creating the connection or edit it later.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
3. On the Data Configuration page, click **Manage Connections** under **Configurations**.
4. On the Manage Connections page, click **Create** and then click **Connection**.
5. In Create Connection, select the applicable option in Usage Type, and then click the connection type you want to create.



6. In the dialog box for the connection, enter the details for your connection in the fields.
7. Click **Save**.

The new connection is available on the Connections page.

Update a Data Connection

When you first make a data connection, or when you make changes, you need to initialize and refresh it.

1. Sign in to the Oracle Cloud Infrastructure Console.
2. In Oracle Cloud Infrastructure Console, click the Navigation menu icon in the top left corner.
3. Click **Analytics & AI**. Under Analytics, click **NetSuite Analytics Warehouse**.
4. Navigate to your service instances page.
5. On the Instances page, click the instance for which you want to update the service.
6. Click **Connections**, then select or search for the connection you want to test.
7. Click the **Action** menu for the connection and select **Initialize/Refresh Connection**.

Test a Data Connection

After you create a data connection, you should test it to ensure it works properly.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
3. On the Data Configuration page, click **Manage Connections** under **Configurations**.
4. On the Manage Connections page, click **Connections**, then select or search for the connection you want to test.
5. Click the **Action** menu for the connection and select **Test Connection**.
6. On the Request History page, check the status of the request to test the connection.

Delete a Data Connection

You can delete a custom data connection if you don't need it anymore.

Ensure that you delete the functional areas, data augmentations, and custom data configurations related to the data connection before deleting it. You can't update or load data from deleted data connections to the warehouse.

> **Note:**

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
3. On the Data Configuration page, click **Manage Connections** under **Configurations**.

4. On the Manage Connections page, click **Connections**, then select or search for the connection you want to test.
5. Click the **Action** menu for the connection and select **Delete**.
6. In the Delete Connection dialog box, click **Delete**.

Connect With On-premises Sources

Connect with your on-premises applications to provide the background information for reports.

You can blend the additional data from these data sources with the prebuilt datasets to enhance business analysis.

Set up the Remote Agent to Load Data into NetSuite Analytics Warehouse (Preview)

As a service administrator, you can use an extract service remote agent to connect to your on-premises systems such as E-Business Suite, Peoplesoft, and JD Edwards, load data from these on-premises systems into Oracle NetSuite Analytics Warehouse, and then use the on-premises data to create data augmentations.

After connecting to your on-premises system, the remote agent extracts the data and loads it into the autonomous data warehouse associated with your Oracle NetSuite Analytics Warehouse instance. You can extract and load the on-premises data into Oracle NetSuite Analytics Warehouse only once in 24 hours.

> **Note:**
>
> After configuring the remote agent on the Data Configuration page, wait for few minutes, refresh the remote agent page, and when you see the Agent Initialised message, you can proceed with other operations such as testing the connection to the remote agent, testng the connection to the remote source like EBusiness Suite, and refreshing the metadata. This enables you to run these jobs without timeout failures because data pipeline has a default timeout of 15 minutes.

Ensure that **Remote Agent** is enabled on the Enable Features page prior to creating this connection. See Make Preview Features Available.

1. Download the remote agent Docker image from here.
2. Identify a host to deploy the remote agent.

   The host that you identify must meet these minimum system requirements for the basic configuration of a single source agent:

   - CPU: 4 (CORE/CPU)
   - Memory: 8 GB
   - Storage: 8 GB

   > **Note:**
   >
   > Ensure that the host is able to make a JDBC connection to the applicable database.

3. Depending on your host, complete these steps:

- For a Linux machine, do the following:

a. Copy the Docker image to the host and load it using this script:

```
docker load -i <docker image zip>
//List the images docker images
```

b. Create and run the Docker container using this script:

```
docker run -d -p 9091:9091 --name remoteagent -v /faw/software/
remoteagent/config/:/faw/software/remoteagent/config/ -v /faw/logs/
RemoteAgent/:/faw/logs/RemoteAgent <docker image Id>
```

If the remote agent user interface isn't accessible, then run this script:

```
sudo docker run -d -p 9091:9091 --name remoteagent --network host -
v /faw/software/remoteagent/config/:/faw/software/remoteagent/config/ -
v /faw/logs/RemoteAgent/:/faw/logs/RemoteAgent <docker image Id>
```

> **Note:**
>
> Ensure that the logs directory in `/faw/logs/RemoteAgent/` has write permissions and the `config` folder in `/faw/software/remoteagent/config/` is present in case you need to add custom properties.

c. Verify that the container has started successfully using this script:

```
run '$ docker ps'
```

d. Configure the extract service URL to connect using this information:

i. Sign in to the remote agent user interface using `https://<host>:9091/extractservice-remoteagent/index.html`.

ii. Configure the extract service URL that the remote agent connects to and configure any outgoing proxies if required using the applicable extract service end points. You can form the extract service url based on your Oracle NetSuite Analytics Warehouse URL by replacing `ui/oax/` with the extract service context path. For example, if your product URL is `https://myinstance.example.com/ui/oax/` then the extract service URL would be `https://myinstance.example.com/extractservice`.

e. In the remote agent user interface, click **Configure** to configure the agent.

f. Copy the configuration details from the text box or download the configuration details.

You use it to set up the connection on the Data Configuration page in Oracle NetSuite Analytics Warehouse.

g. Optional: If you need to upgrade the remote agent in the Linux host, then use the following script:

```
Stop Remoteagent docker
sudo docker stop remoteagent
```

```
Remove Remoteagent docker
sudo docker rm remoteagent

load the image from tar ball
docker load -i <docker image zip>

Get Image ID from below
sudo docker images

Run the image:
sudo docker run -d -p 9091:9091 --name remoteagent -v /faw/software/
remoteagent/config/:/faw/software/remoteagent/config/:Z -v /faw/logs/
RemoteAgent/:/faw/logs/RemoteAgent:Z <imageid>
```

- For a Windows machine, do the following:

a. Load the docker using this script:

```
docker load -i .\RemoteAgentFrameworkDocker_19.5.tar.gz
```

Ensure to replace "19.5" with the latest docker version number in the script.

b. Provide permission to these directories

```
icacls "C:/faw/software/remoteagent/" /grant Everyone:F /t
         icacls "C:/faw/logs/RemoteAgent/" /grant Everyone:F /t
```

> **Note:**
>
> - icacls: This is a command-line tool used to manage file and directory access control lists (ACLs).
> - C:/faw/software/remoteagent/": This specifies the target directory where ACL changes will be applied.
> - /grant Everyone:F: This grants the "Everyone" group Full Control (F) permissions to the specified directory and all subdirectories and files recursively due to the "/t" switch. The user needs write permissions, hence "F" (full control was given). To use user-specific permission, replace "Everyone" with username.

c. Run the docker using this script:

```
docker run -d -p 9091:9091 --name remoteagent -v C:/faw/software/
remoteagent/config/:/faw/software/remoteagent/config/
      -v C:/faw/logs/RemoteAgent/:/faw/logs/RemoteAgent <imageid>
```

4. You can either use the default TLS certificate provided in the keystore with the remote agent or provide your own keystore and TLS certificate.

   To provide your own keystore and TLS certificate, complete these steps:

a. If you don't have a keystore, then generate one using the keytool CLI command such as:

```
keytool -genkeypair -alias springboot -keyalg RSA -keysize 4096 -
storetype PKCS12 -keystore springboot.p12 -validity 3650 -storepass
password -ext SAN=dns:test.example.com
```

Instructions on how to use the keytool CLI command can be found here.

b. After generating the keystore, place it in the `/faw/software/remoteagent/config` directory of your local instance. Once in this directory, you must create a `startup-config.properties` file. This properties file contains the keystore information and credentials needed by the remote agent to connect to it and fetch the TLS certificate.

c. In the `startup-config.properties` file, add the following key-value properties:

```
server.ssl.enabled=true
server.ssl.key-store=</PATH/TO/KEYSTORE_FILE>
server.ssl.key-store-password=<KEYSTORE_PASSWORD>
server.ssl.key-store-type=<KEYSTORE_TYPE>
server.ssl.key-alias=<KEYSTORE_ALIAS>
server.ssl.key-password=<KEY_PASSWORD>
```

The values for each key pair are as follows:

- </PATH/TO/KEYSTORE_FILE> - File location of the keystore file
- <KEYSTORE_PASSWORD> - Password specified for the keystore
- <KEYSTORE_TYPE> - Type specified for the keystore, should be either JKS or PKCS12
- <KEYSTORE_ALIAS> - Alias specified for the keystore
- <KEY_PASSWORD> - Certificate password, NOT the password for the keystore

d. After specifying all the required properties in the startup-config.properties file, restart the remote agent docker. The remote agent uses your TLS certificate instead of the default.

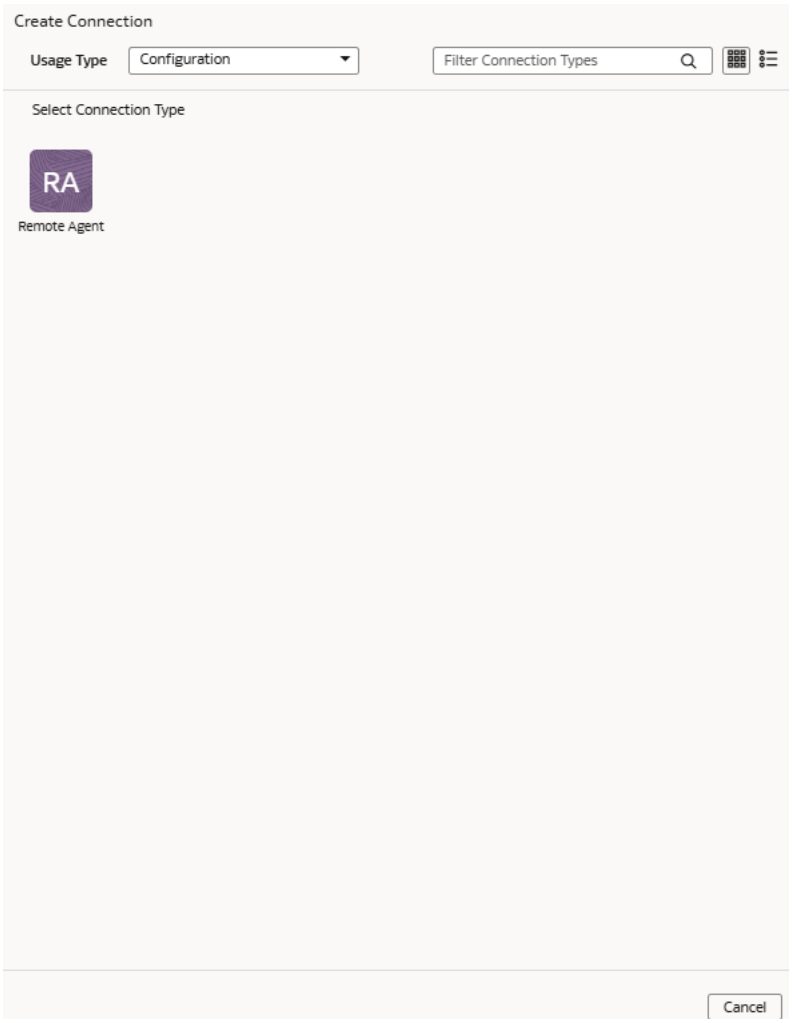
5. Configure the remote agent on the Data Configuration page in Oracle NetSuite Analytics Warehouse using these instructions:

a. On the **Console**, click **Data Configuration** under **Application Administration**.

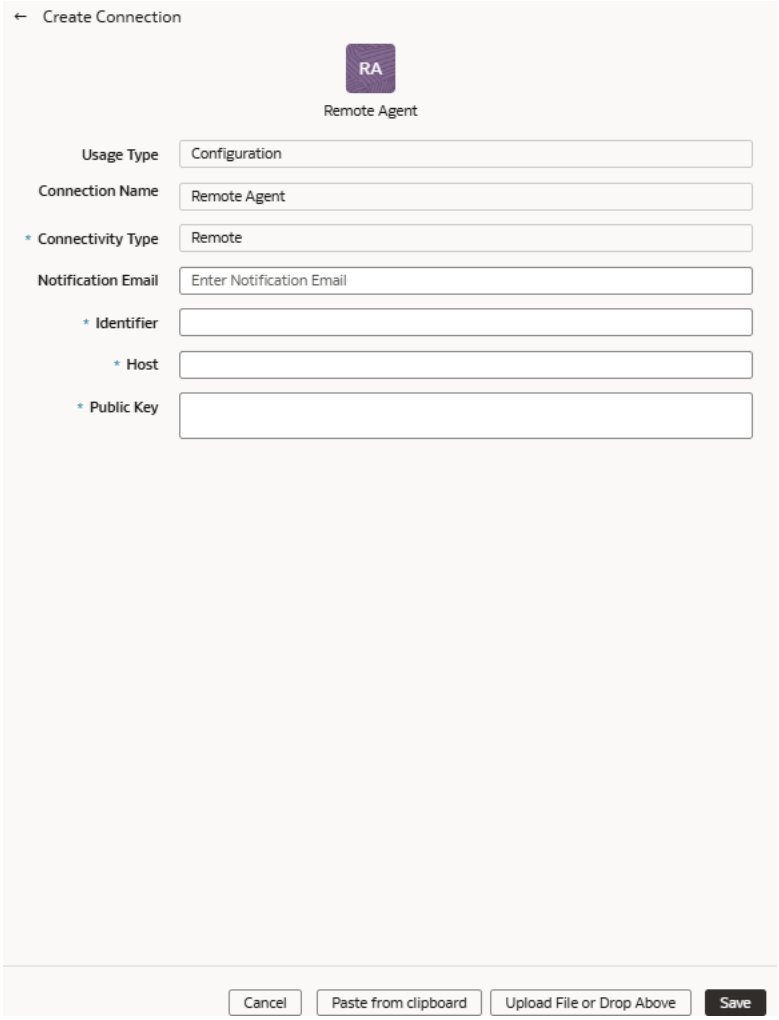
b. On the Data Configuration page, click **Manage Connections**.

c. On the Manage Connections page, click **Create** and then click **Connection**.

d. In Create Connection, select **Configuration** in **Usage Type**, and then select **Remote Agent** as the connection type.



e. In the Create Connection Remote Agent dialog, in **Connection Name**, you can modify the default name and verify that **Remote** is displayed in **Connectivity Type**.



f. Enter an email address to receive notifications in **Notification Email**, provide the **Identifier** and **Host**, in **Public Key**, click **Upload File or Drop Above** to fill in the details of the remote agent, and then click **Save**. You can add the configuration details file that you had downloaded or use the configuration details that you had copied after configuring the remote agent.

Load Data from On-premises E-Business Suite into NetSuite Analytics Warehouse (Preview)

As a service administrator, you can use an extract service remote agent to connect to your on-premises Oracle E-Business Suite system.

After connecting to your on-premises system, the remote agent extracts the data and loads it into the autonomous data warehouse associated with your Oracle NetSuite Analytics Warehouse instance. The remote agent pulls the metadata through the public extract service REST API and pushes data into object storage using the object storage REST API. You can extract and load the on-premises data into Oracle NetSuite Analytics Warehouse only once a day. Ensure that the user credentials you provide have access to the specific tables they need to extract data from within the EBS schema, whose URL you provide while creating the connection.

Ensure that **Oracle E-Business Suite On-Prem** is enabled on the Enable Features page prior to creating this connection. See Make Preview Features Available.

1. Set up the remote agent to load data from your on-premises Oracle E-Business Suite system into Oracle NetSuite Analytics Warehouse.

   See Set up the Remote Agent to Load Data into NetSuite Analytics Warehouse (Preview).

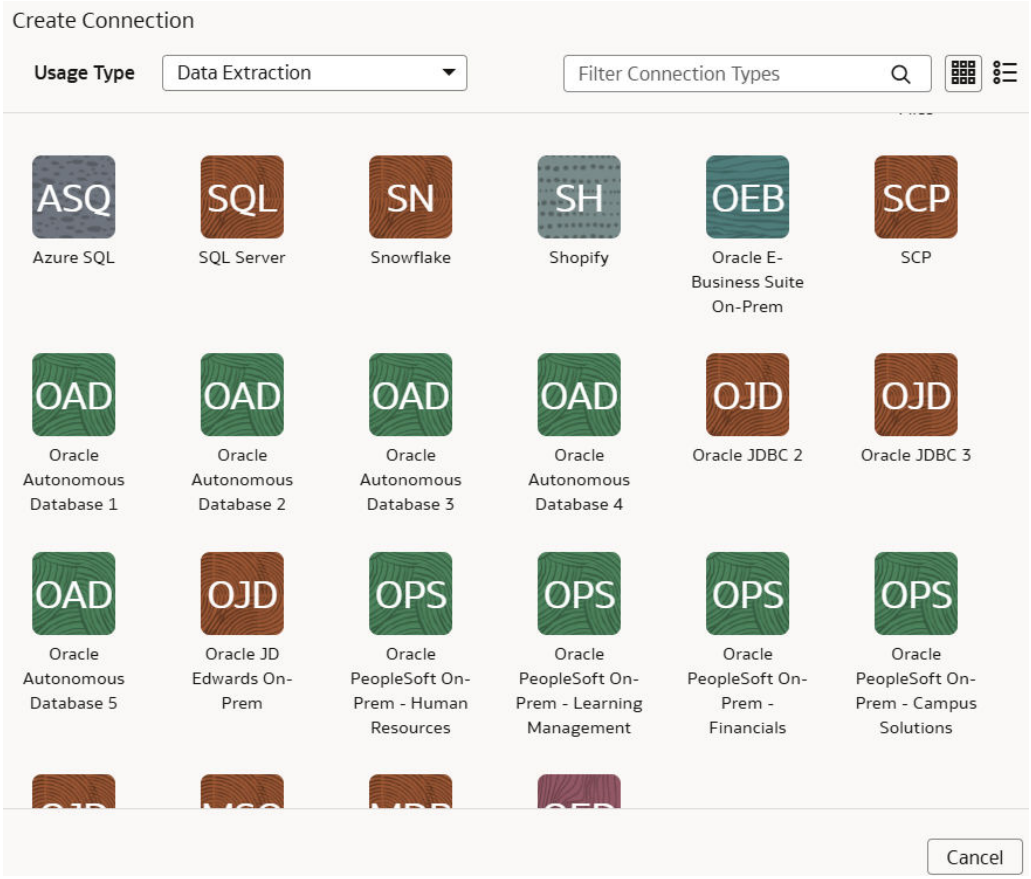
2. Configure the remote agent and E-Business Suite data source on the Data Configuration page in Oracle NetSuite Analytics Warehouse using these instructions:

   a. On the **Console**, click **Data Configuration** under **Application Administration**.

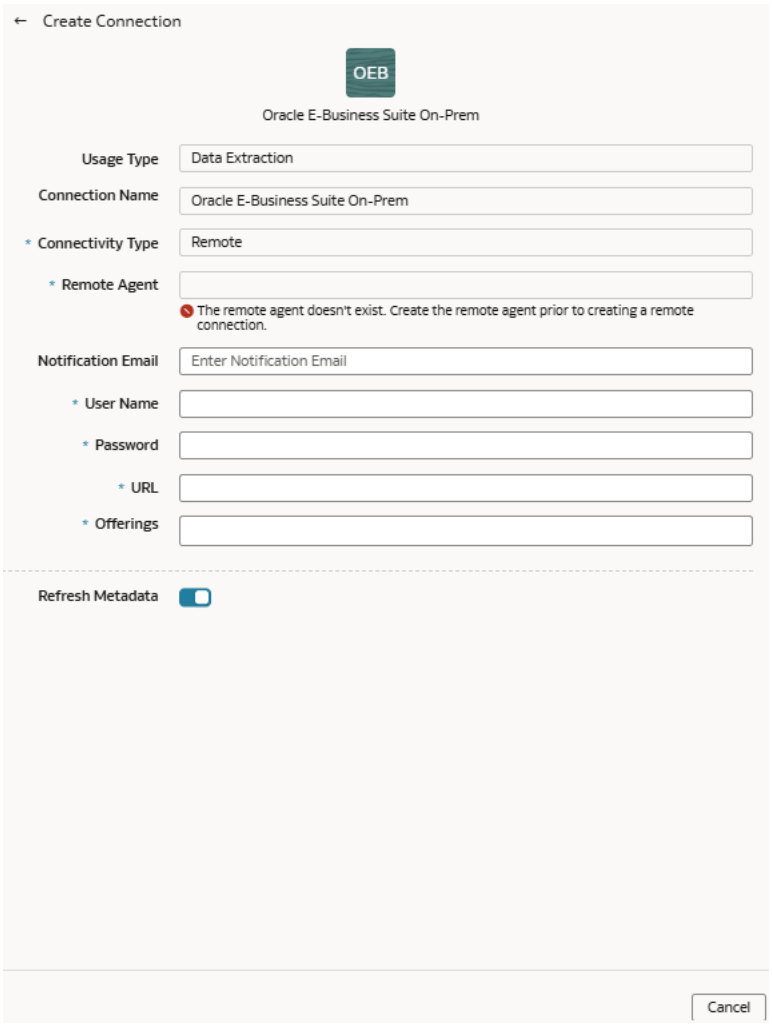
   b. On the Data Configuration page, click **Manage Connections**.

   c. On the Manage Connections page, click **Create** and then click **Connection**.

   d. In Create Connection, select **Data Extraction** in **Usage Type**, and select **Oracle E-Business Suite On-Prem** as the connection type.



   e. In Create Connection for Oracle E-Business Suite On-Prem, select **Remote** as connectivity type.



f. In the **Remote Agent** field, select the remote agent connection that you created, for example, **EBS-Remote Agent**. Enter an email address to receive notifications in **Notification Email**, provide the credentials in **User Name** and **Password**, the E-Business Suite connection using the JDBC format such as `jdbc:oracle:thin:@<HOST>:<PORT>/<DB_NAME/SID_NAME>` in **URL**, and select **Application Object Library** and **General Ledger** offerings mandatorily, and any other E-Business Suite offerings that you want to load data from in **Offerings**.

g. Verify that the **Refresh Metadata** toggle is enabled to ensure that the metadata is refreshed when you save the connection. You can later refresh the metadata from the Actions menu on the Manage Connections page, if required.

> **Note:**
>
> You can't create augmentations for E-Business Suite On-Prem unless you perform a metadata extract.

h. Confirm that you see the Remote Agent and E-Business Suite connections on the Manage Connections page.

i. Test both the connections by selecting the **Test Connection** option in **Actions**. You can check the statuses of all these requests on the Data Configuration Request History page.

3. After the connections are successfully established, navigate to the Data Configuration page, select the connection in **Data Source**, then click the Data Augmentation tile, and create a data augmentation using the E-Business Suite data. Select the applicable E-Business Suite source tables. See Augment Your Data.

Load Data from On-premises JD Edwards into NetSuite Analytics Warehouse (Preview)

As a service administrator, you can use an extract service remote agent to connect to your on-premises JD Edwards system and use the JD Edwards data to create data augmentations.

After connecting to your on-premises system, the remote agent extracts the data and loads it into the autonomous data warehouse associated with your Oracle NetSuite Analytics Warehouse instance. The remote agent pulls the metadata through the public extract service REST API and pushes data into object storage using the object storage REST API. You can extract and load the on-premises data into Oracle NetSuite Analytics Warehouse only once in 24 hours.
Ensure that **Remote Agent** and **Oracle JD Edwards On-Prem** are enabled on the Enable Features page prior to creating this connection. See Make Preview Features Available.

1. Set up the remote agent to load data from your on-premises Oracle JD Edwards system into Oracle NetSuite Analytics Warehouse.

   See Set up the Remote Agent to Load Data into NetSuite Analytics Warehouse (Preview).

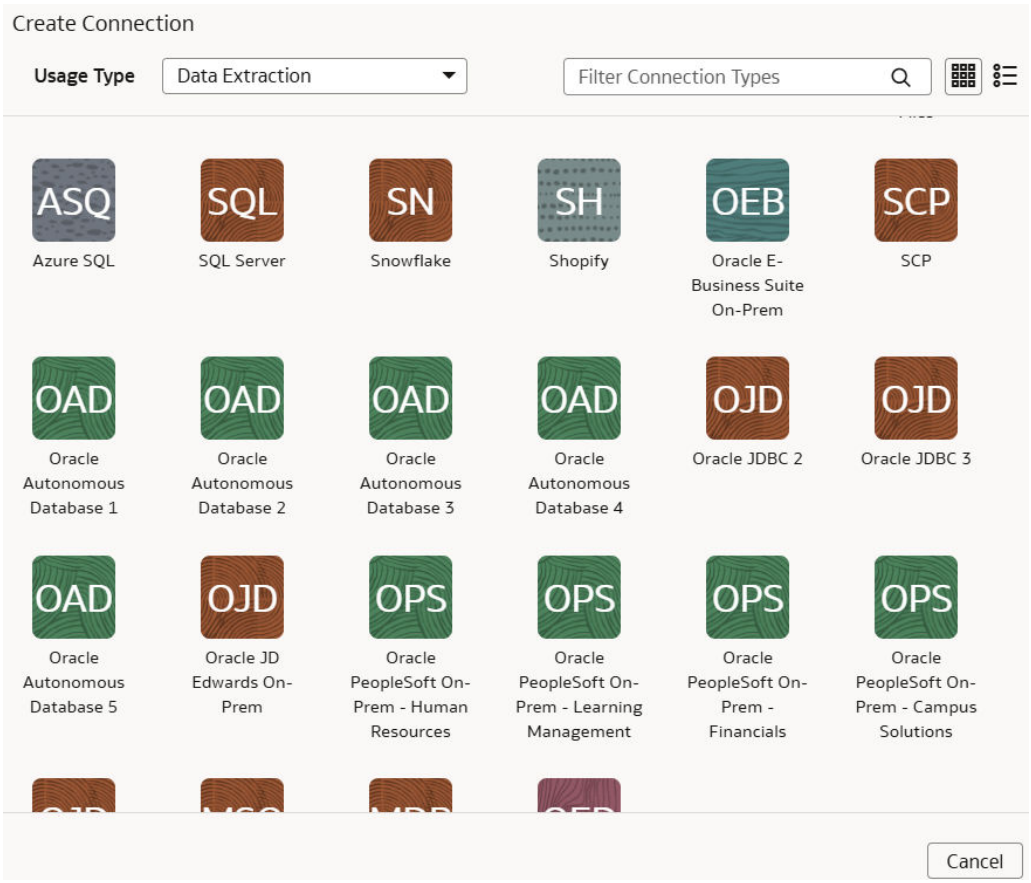
2. Configure the remote agent and JD Edwards data source on the Data Configuration page in Oracle NetSuite Analytics Warehouse using these instructions:

   a. On the **Console**, click **Data Configuration** under **Application Administration**.

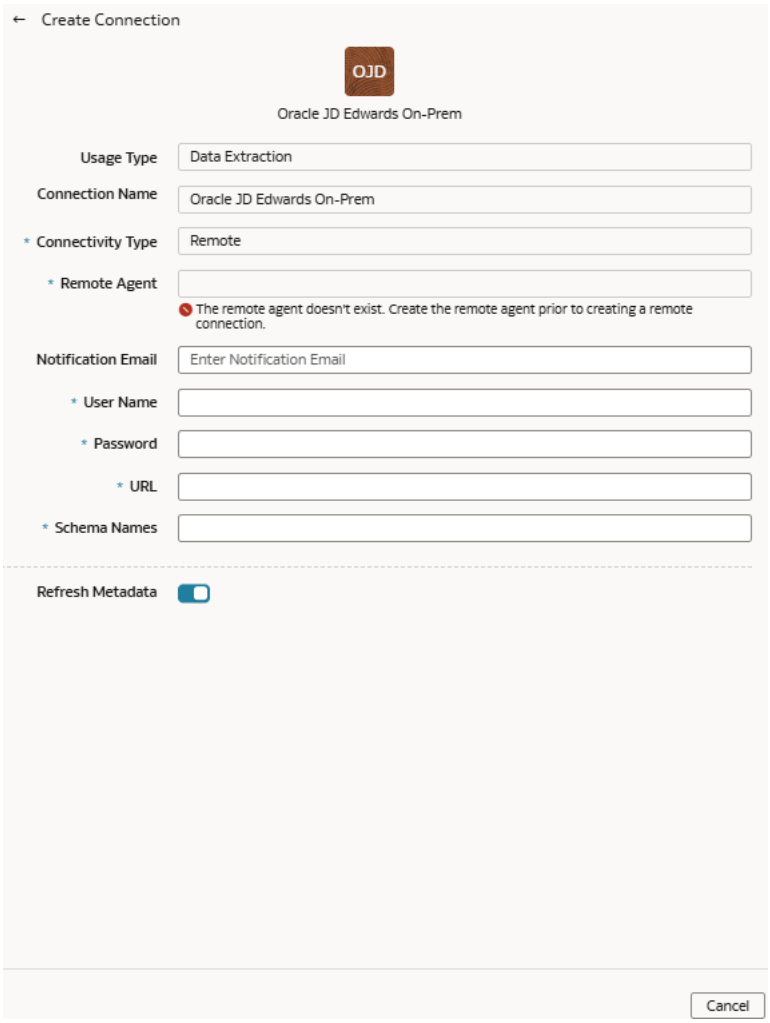
   b. On the Data Configuration page, click **Manage Connections**.

   c. On the Manage Connections page, click **Create** and then click **Connection**.

   d. In Create Connection, select **Data Extraction** in **Usage Type**, and then select **Oracle JD Edwards On-Prem** as the connection type.



e. In Create Connection for Oracle JD Edwards On-Prem, in **Connectivity Type**, verify that **Remote** is selected automatically.



f. In **Remote Agent**, select the remote agent connection that you created earlier, for example, **Remote Agent**.

g. Enter an email address to receive notifications in **Notification Email**, provide credentials for your JD Edwards source in **User Name** and **Password**, and the URL of your JD Edwards source in **URL**.

h. Verify that the **Refresh Metadata** toggle is enabled to ensure that the metadata is refreshed when you save the connection. You can later refresh the metadata from the Actions menu on the Manage Connections page, if required.

> **Note:**
>
> You can't create augmentations for JD Edwards unless you perform a metadata extract.

i. Confirm that you see the Remote Agent and JD Edwards connections on the Manage Connections page.

j. Test both the connections by selecting the **Test Connection** option in **Actions**. You can check the statuses of all these requests on the Data Configuration Request History page.

3. After the connections are successfully established, navigate to the Data Configuration page, select the connection in **Data Source**, then click the Data Augmentation tile, and create a data augmentation using the JD Edwards data. Select the applicable JD Edwards source tables. Select the applicable JD Edwards source tables. See Augment Your Data.

Load Data from On-premises PeopleSoft into NetSuite Analytics Warehouse (Preview)

As a service administrator, you can use an extract service remote agent to connect to your on-premises Oracle PeopleSoft system.

After connecting to your on-premises system, the remote agent extracts the data and loads it into the autonomous data warehouse associated with your Oracle NetSuite Analytics Warehouse instance. The remote agent pulls the metadata through the public extract service REST API and pushes data into object storage using the object storage REST API. You can extract and load the on-premises data into Oracle NetSuite Analytics Warehouse only once in 24 hours.
Ensure that **Remote Agent** and depending on the functional module you want to connect to, the applicable feature is enabled on the Enable Features page prior to creating this connection:

- **Oracle PeopleSoft On-Prem - Campus Solutions**
- **Oracle PeopleSoft On-Prem - Financials**
- **Oracle PeopleSoft On-Prem - Human Resources**
- **Oracle PeopleSoft On-Prem - Learning Management**

See Make Preview Features Available.

1. Set up the remote agent to load data from your on-premises Oracle E-Business Suite system into Oracle NetSuite Analytics Warehouse.

   See Set up the Remote Agent to Load Data into NetSuite Analytics Warehouse (Preview).

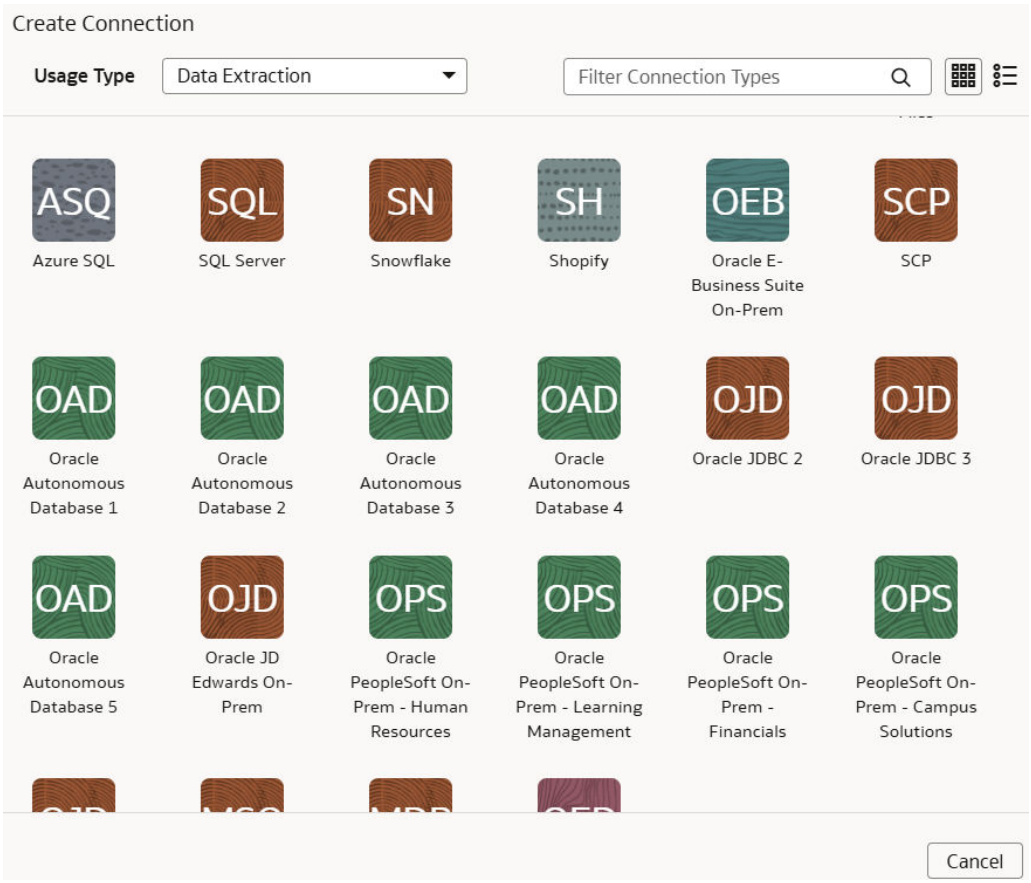
2. Configure the remote agent and PeopleSoft data source on the Data Configuration page in Oracle NetSuite Analytics Warehouse using these instructions:

   a. On the **Console**, click **Data Configuration** under **Application Administration**.

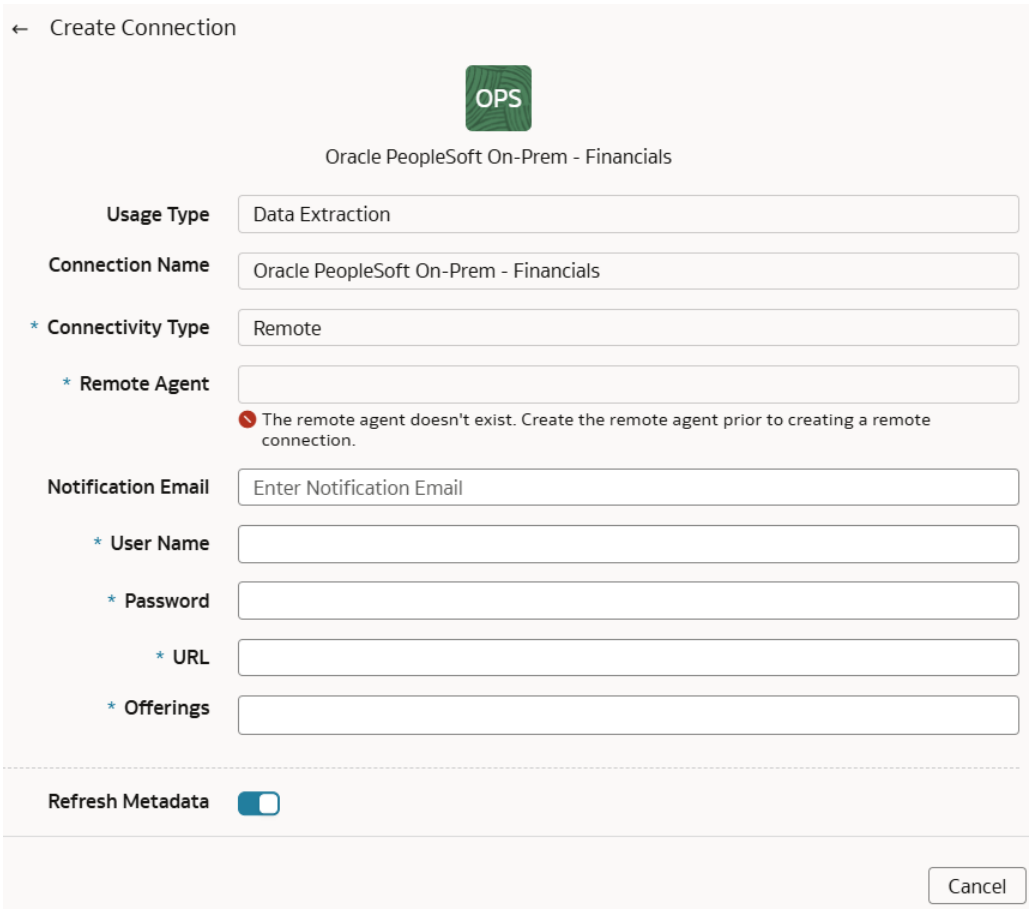
   b. On the Data Configuration page, click **Manage Connections**.

   c. On the Manage Connections page, click **Create** and then click **Connection**.

   d. In Create Connection, select **Data Extraction** in **Usage Type**, and then select the connection type based on the functional module that you want to connect to. For example, to connect to the "Financials" module, select **Oracle PeopleSoft On-Prem - Financials** as the connection type.



e. In Create Connection for Oracle PeopleSoft On-Prem - Financials dialog, in **Connectivity Type**, verify that **Remote** is selected automatically.



f. In **Remote Agent**, select the remote agent connection that you created earlier, for example, **Remote Agent**.

g. Enter an email address to receive notifications in **Notification Email**, provide credentials for your PeopleSoft source in **User Name** and **Password**, and the URL of your PeopleSoft source in **URL**.

h. Verify that the **Refresh Metadata** toggle is enabled to ensure that the metadata is refreshed when you save the connection. You can later refresh the metadata from the Actions menu on the Manage Connections page, if required.

> **Note:**
>
> You can't create augmentations for PeopleSoft unless you perform a metadata extract.

i. Click **Save**.

j. On the Manage Connections page, select **Actions** for the PeopleSoft connection and then select **Test Connection**. You can check the statuses of all these requests on the Data Configuration Request History page.

3. After the connections are successfully established, navigate to the Data Configuration page, select the connection in **Data Source**, then click the Data Augmentation tile, and create a data augmentation using the PeopleSoft data. Select the applicable PeopleSoft source tables. See Augment Your Data.

Load Data from On-premises MySQL Database into NetSuite Analytics Warehouse (Preview)

As a service administrator, you can use an extract service remote agent to connect to your on-premises MySQL database.

After connecting to your on-premises system, the remote agent extracts the data and loads it into the autonomous data warehouse associated with your Oracle NetSuite Analytics Warehouse instance. The remote agent pulls the metadata through the public extract service REST API and pushes data into object storage using the object storage REST API. You can extract and load the on-premises data into Oracle NetSuite Analytics Warehouse only once a day.
Ensure that **MySQL On-Prem** is enabled on the Enable Features page prior to creating this connection. See Make Preview Features Available.

1. Set up the remote agent to load data from your on-premises MySQL database into Oracle NetSuite Analytics Warehouse.

   See Set up the Remote Agent to Load Data into NetSuite Analytics Warehouse (Preview).

2. Check the port number for your on-premises MySQL database and create a service request with server host and port details to enable network connectivity to the on-premises MySQL server.

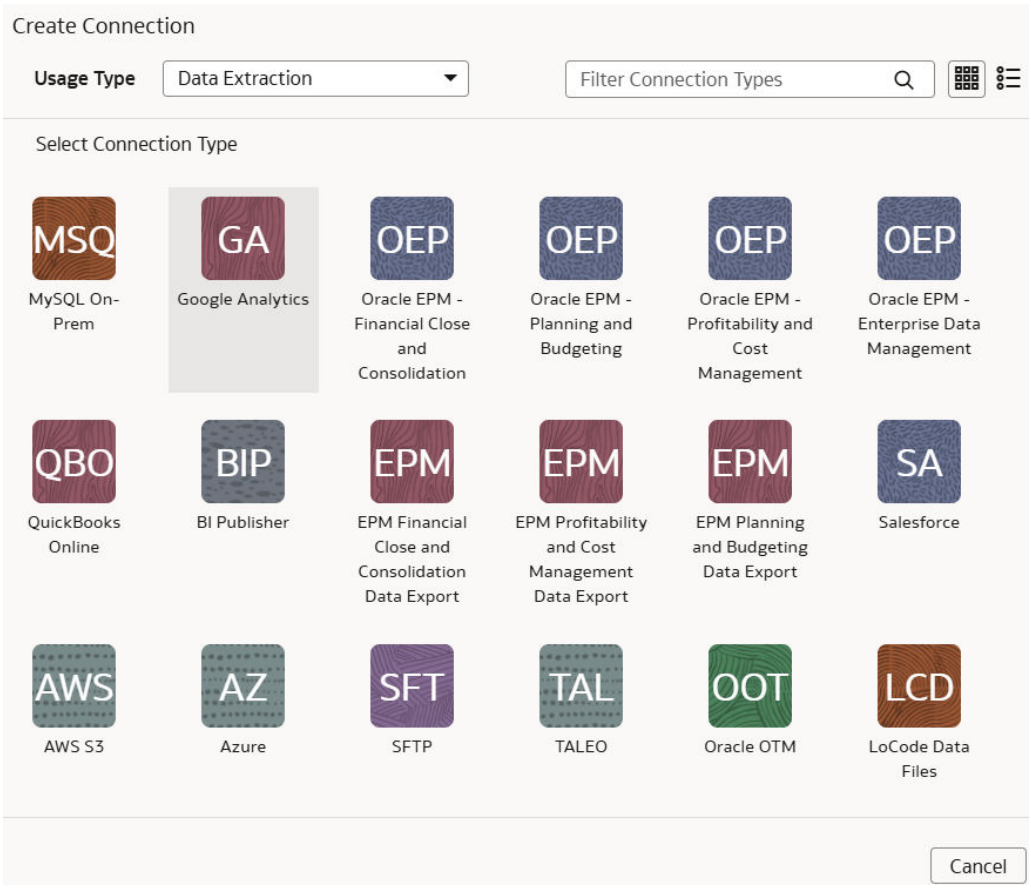
3. Specify the remote agent and configure the on-premises MySQL database on the Data Configuration page in Oracle NetSuite Analytics Warehouse by following these steps:

   a. On the **Console**, click **Data Configuration** under **Application Administration**.

   b. On the Data Configuration page, click **Manage Connections**.

   c. On the Manage Connections page, click **Create** and then click **Connection**.

   d. In Create Connection, select **Data Extraction** in **Usage Type**, and then select **MySQL On-Prem** as the connection type.



e. In **Connectivity Type**, select **Remote**.

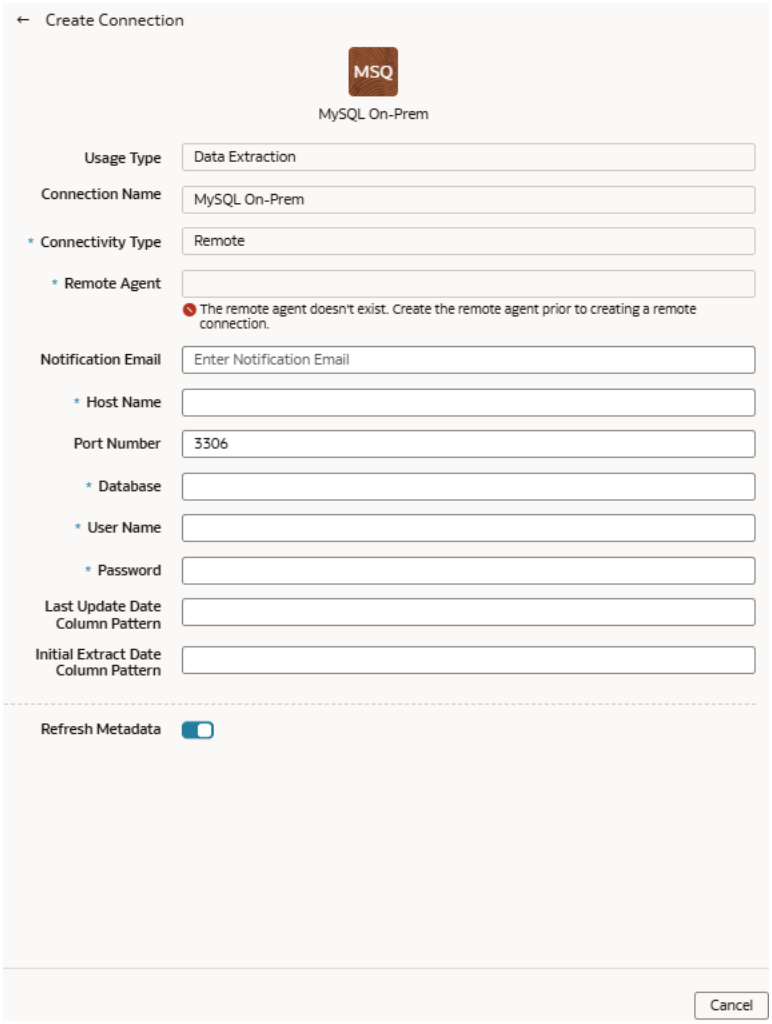
f. In the **Remote Agent** field, select the remote agent connection that you created, for example, **MySQL-Remote Agent**. Enter an email address to receive notifications in **Notification Email**, and provide these details:

- **Host Name**: Enter the host name of MySQL server such as `100.111.252.64`
- **Port Number**: Enter the port number where the server is listening such as `3306`
- **Database**: Enter the database name you need to connect to such as `airportdb`
- Credentials to access the database in **User Name** and **Password**
- **Last Update Date Column Pattern**: Enter `"%r%a%o%%"`

> **Note:**
>
> If column name matches with pattern provided, then NetSuite Analytics Warehouse marks the column name with `isLastUpdateDate = true` and uses it for the incremental extract. For example, if pattern provided is `"%mo%fie%te%"`, then the column name `modifiedDate` is marked as `isLastUpdateDate = true`.

- **Initial Extract Date Column Pattern**: Enter `"%e%art%new"`

> **Note:**
>
> If column name matches with pattern provided, then NetSuite Analytics Warehouse marks the column name with `isCreationDate = true` and uses it for the initial extract date extraction. For example, if pattern provided is: `"%cr%ted%te%"`, then the column name `createdDate` is marked as `isCreationDate = true`.



g. Verify that the **Refresh Metadata** toggle is enabled to ensure that the metadata is refreshed when you save the connection. You can later refresh the metadata from the Actions menu on the Manage Connections page, if required.

> **Note:**
>
> You can't create augmentations for on-premises MySQL database unless you perform a metadata extract.

h. Confirm that you see the Remote Agent and on-premises MySQL database connections on the Manage Connections page.

i. Test both the connections by selecting the **Test Connection** option in **Actions**. You can check the statuses of all these requests on the Data Configuration Request History page.

4. After the connections are successfully established, navigate to the Data Configuration page, select the connection in **Data Source**, then click the Data Augmentation tile, and create a data augmentation using the on-premises MySQL database data. Select the applicable on-premises MySQL database source tables. See Augment Your Data.

Load Data from SQL Server into Oracle NetSuite Analytics Warehouse (Preview)

As a service administrator, you can use the Oracle NetSuite Analytics Warehouse extract service to acquire data from SQL Server and use it to create data augmentations.

Ensure that **SQL Server** is enabled on the Enable Features page prior to creating this connection. See Make Preview Features Available.

1. Set up the remote agent to load data from your SQL Server into Oracle NetSuite Analytics Warehouse.

   See Set up the Remote Agent to Load Data into NetSuite Analytics Warehouse (Preview).

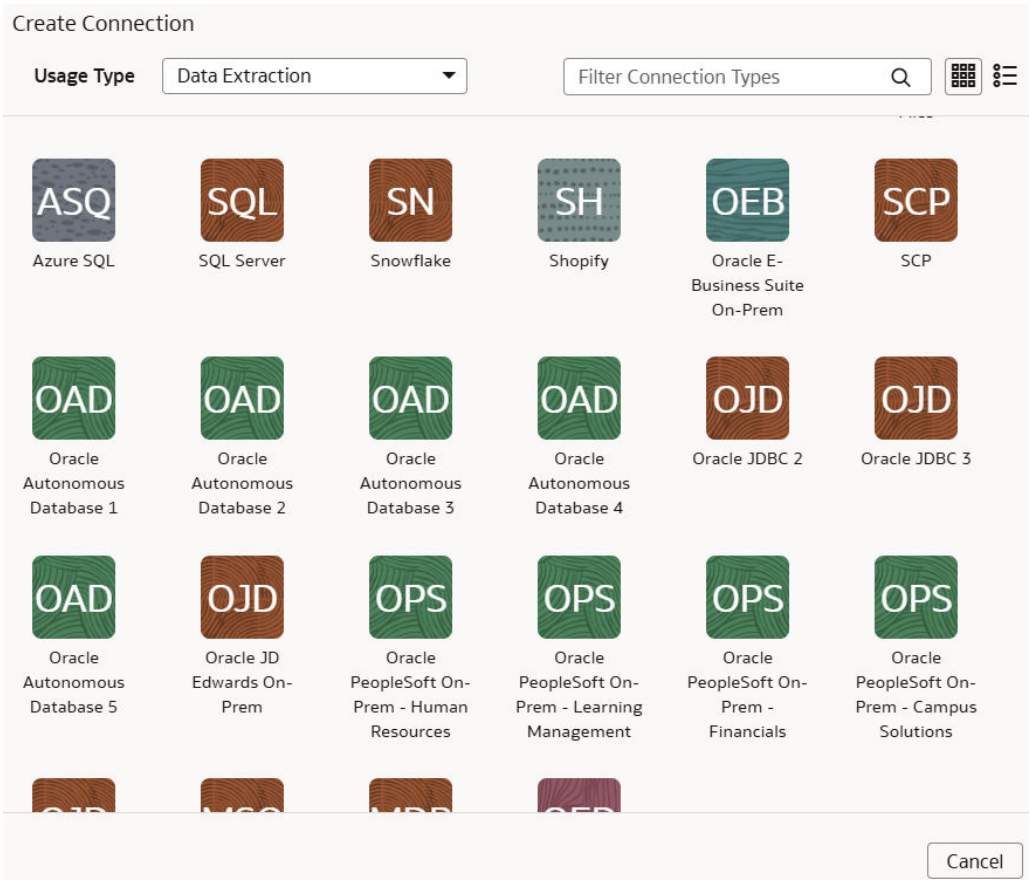
2. In Oracle NetSuite Analytics Warehouse, create the SQL Server data connection using these instructions:

   a. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.

   b. On the Data Configuration page, click **Manage Connections**.

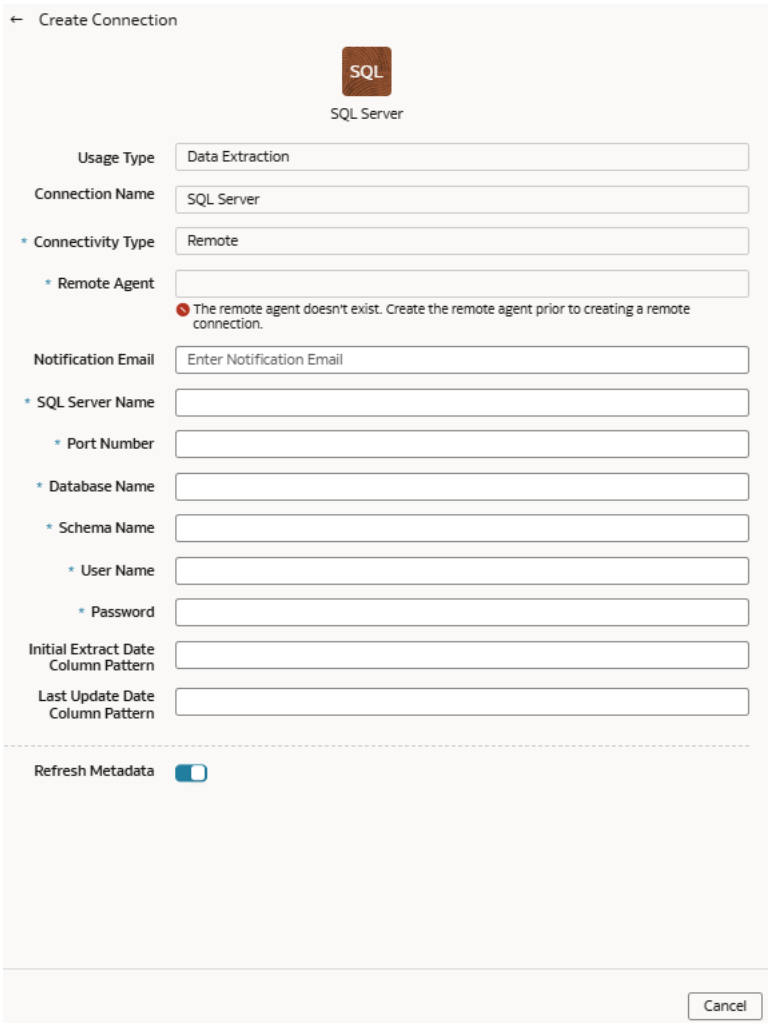
   c. On the Manage Connections page, click **Create** and then click **Connection**.

   d. In Create Connection, select **Data Extraction** in **Usage Type**, and then select **SQL Server** as the connection type.



e. In the dialog for the SQL Server connection, enter these details and click **Save**:

- **Connectivity Type**: Select **Remote**.
- **Remote Agent**: Select the remote agent that you had set up to load data from your SQL Server.
- **Notification Email**: Enter an email address to receive notifications.
- **SQL Server Name**: Enter the SQL server name.
- **Port Number**: Enter the port number on which your SQL server is available.
- **Database Name**: Enter the database name in your SQLServer instance.
- **Schema Name**: Enter the name of the schema for the dataset you want to load to run analytics on.
- **User Name** and **Password**: Enter the credentials for your SQL Server instance.
- **Initial Extract Date Column Pattern**: MM/dd/yyyy is the date format in your initial extract column; sample date is 1/23/1998.
- **Last Update Date Column Pattern**: Last update date shows when the last update was done in your SQL Server database.



f. Verify that the **Refresh Metadata** toggle is enabled to ensure that the metadata is refreshed when you save the connection.

   You can later refresh the metadata from the Actions menu on the Manage Connections page, if required.

> **Note:**
>
> You can't create augmentations for SQL Server unless you perform a metadata extract.

g. Click **Save**.

3. On the Manage Connections page, select **Actions** for the SQL Server connection and then select **Test Connection**.
4. After the connections are successfully established, navigate to the Data Configuration page, select the connection in **Data Source**, then click the Data Augmentation tile, and create a data augmentation using the SQL Server data. Select the applicable SQL Server source tables. See Augment Your Data.

Connect with Cloud File Storage Sources

Connect with your file storage-based cloud sources to provide the background information for reports.

You can blend the additional data from these data sources with the prebuilt datasets to enhance business analysis. The file-based connectors support only UTF-8 encoding for data files that you upload.

About Date and Timestamp Formatting for CSV File-based Extractors

Extractors such as Secure FTP (SFTP), Amazon Simple Storage Service (AWS S3), and Oracle Object Storage Service use CSV data files that have date and timestamp fields.

For the CSV file-based extractors, use the format examples to provide the values in the CSV Date Format, and CSV Timestamp Format fields while entering the source connection details.

> **Note:**
>
> Ensure that the date and timestamp formats for the data files match the date and timestamp formats in your source; for example, if you've used MM/dd/yyyy and MM/dd/yyyy hh:mm:ss in your source, then you must specify the same formats while creating the applicable data connections.

Examples

Example	Pattern
1/23/1998	MM/dd/yyyy
1/23/1998 12:00:20	MM/dd/yyyy hh:mm:ss
12:08 PM	h:mm a
01-Jan-1998	dd-MMM-yyyy
2001-07-04T12:08:56.235-0700	yyyy-MM-dd'T'HH:mm:ss.SSSZ

The guidelines to define the format are:

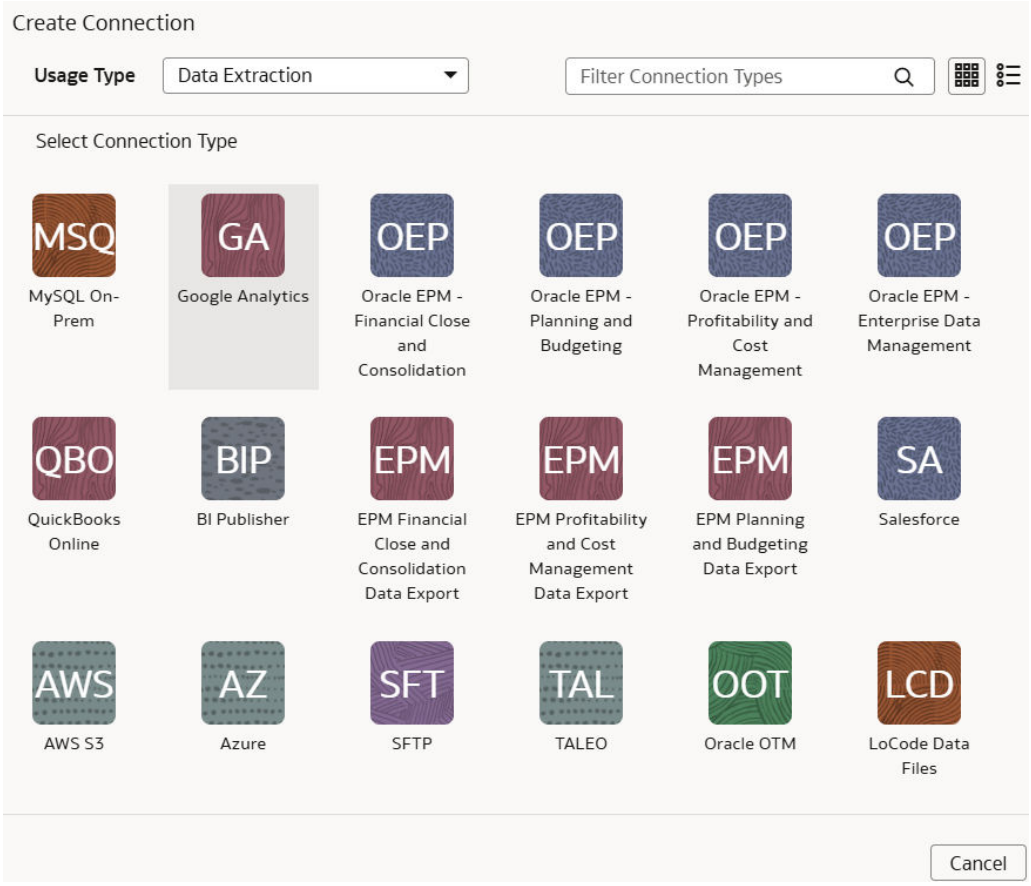
Letter	Meaning
M	Month
d	Day
y	Year
h	Hour (0-12)
H	Hour (0-23)
m	Minute
s	Second
S	Milli Second
a	AM/PM
Z	Timezone

Load Data from Amazon Simple Storage Service into Oracle NetSuite Analytics Warehouse (Preview)

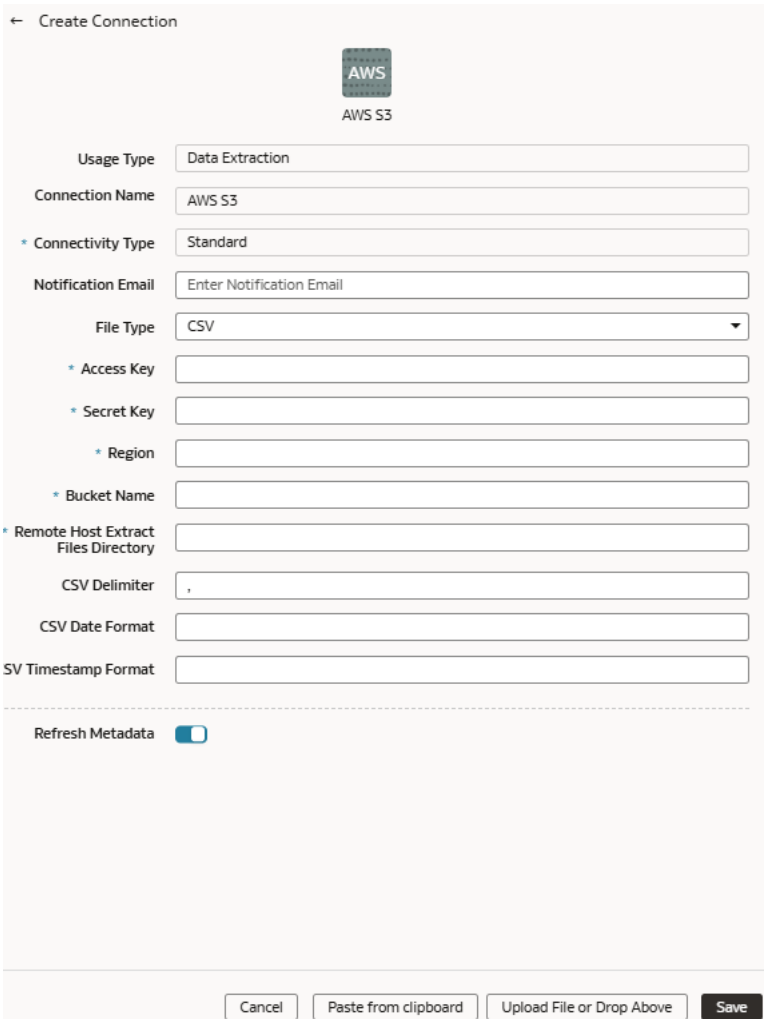
As a service administrator, you can use the Oracle NetSuite Analytics Warehouse extract service to acquire data from Amazon Simple Storage Service (AWS S3) and use it to create data augmentations.

Ensure that **AWS S3** is enabled on the Enable Features page prior to creating this connection. See Make Preview Features Available.

1. In Oracle NetSuite Analytics Warehouse, create the AWS S3 data connection using these instructions:
   a. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
   b. On the Data Configuration page, click **Manage Connections**.
   c. On the Manage Connections page, click **Create** and then click **Connection**.
   d. In Create Connection, select **Data Extraction** in **Usage Type**, and then select **AWS S3** as the connection type.



   e. In the dialog for the AWS S3 connection, select **Standard** in **Connectivity Type**, enter an email address to receive notifications in **Notification Email**, and provide applicable details of your AWS S3.



f. Verify that the **Refresh Metadata** toggle is enabled to ensure that the metadata is refreshed when you save the connection.

   You can later refresh the metadata from the Actions menu on the Manage Connections page, if required.

> **Note:**
>
> You can't create augmentations for AWS S3 unless you perform a metadata extract.

g. Click **Save**.

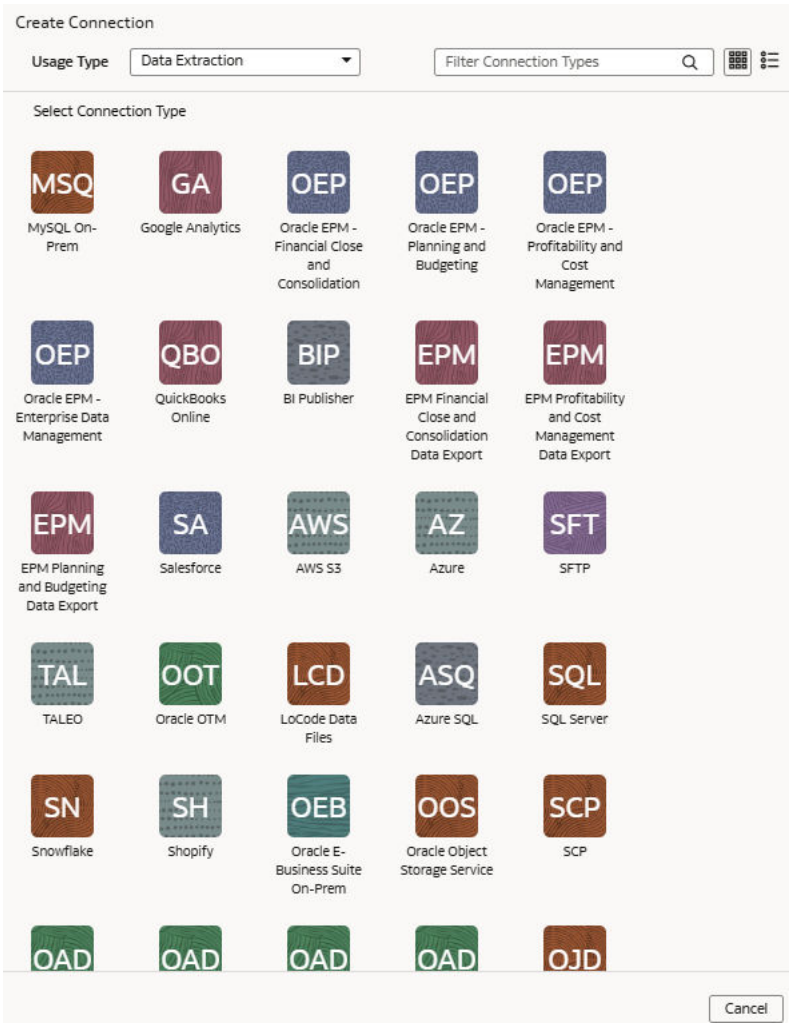
2. On the Manage Connections page, select **Actions** for the AWS S3 connection and then select **Test Connection**.
3. After the connections are successfully established, navigate to the Data Configuration page, select the connection in **Data Source**, then click the Data Augmentation tile, and create a data augmentation using the AWS S3 data. Select the applicable AWS S3 source tables. See Augment Your Data.

Load Data from Oracle Object Storage into NetSuite Analytics Warehouse

As a service administrator, you can use the NetSuite Analytics Warehouse extract service to acquire data from Oracle Object Storage Service and use it to create data augmentations.

The recommended approach is to create one augmentation from one source table after acquiring data from Oracle Object Storage Service. After completion of augmentation, NetSuite Analytics Warehouse renames the source table in this case and if you create more than one augmentation from the same source, all other augmentations may fail with a message that the source file wasn't found.

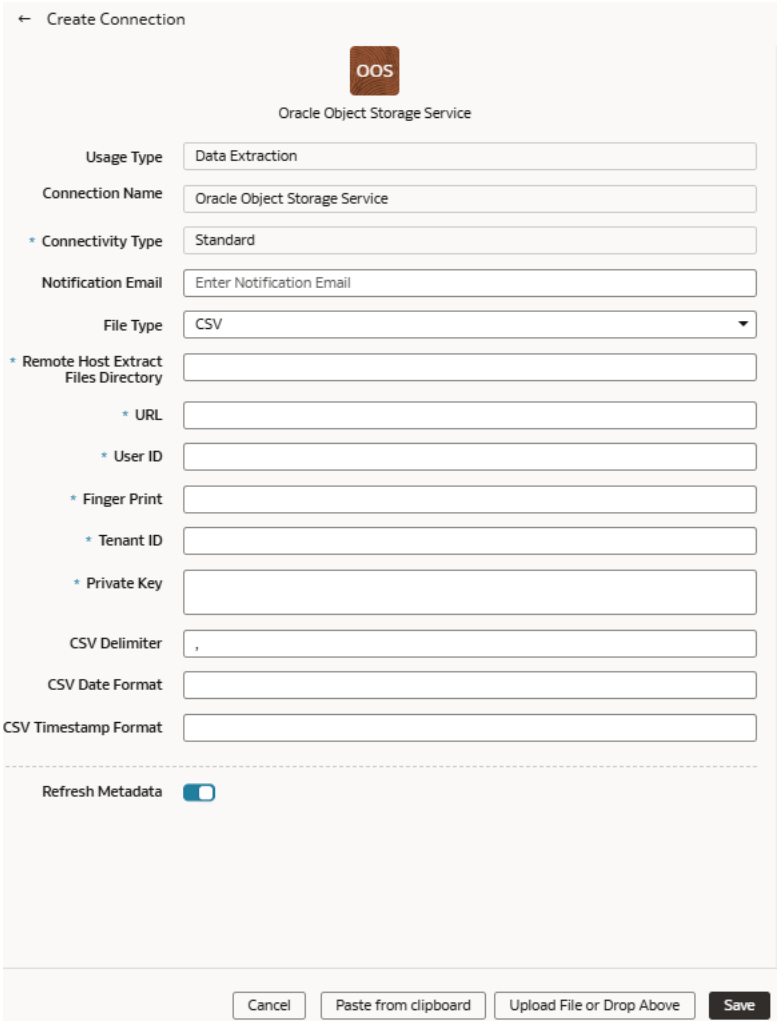
1. Store the following details in a text file to use while creating the connection to Oracle Object Storage Service in NetSuite Analytics Warehouse:
    a. In Oracle Object Storage Service, create the Remote Host Extract Files directory as the base folder in which you must place all your data files. Note down the name of this directory. See the "To create a folder or subfolder" section in Using the Console.
    b. Obtain the URL of the Oracle Object Storage Service by signing into the Oracle Cloud Infrastructure Console and navigating to the bucket to get the details of the region, namespace, and bucket name. For example, the URL must be in the `https://objectstorage.<region>.oraclecloud.com/n/<namespace>/b/<name of the bucket>` format. See the "To view bucket details" section in Using the Console.
    c. Obtain a user's OCID by navigating in the Oracle Cloud Infrastructure Console to **Identity & Security**, and then **Users**. On the Users page, search for a user who has access to the bucket used in the connector and copy the **OCID**. Obtain the tenancy ID by clicking your profile icon and then **Tenancy** in the Oracle Cloud Infrastructure Console. Under Tenancy information, copy the **OCID**. See Where to Get the Tenancy's OCID and User's OCID.
    d. Obtain the fingerprint for a user from the Oracle Cloud Infrastructure Console. Navigate to **API Keys** under **Resources** on the user page, and then click **Add API Keys**. In the Add API Keys dialog, ensure that **Generate API Key Pair** is selected. Download the private and public keys using the **Download Private Key** and **Download Public Key** options. You must copy the entire text of the private key along with the comments before and after the actual key. These comments could be as simple as: "----------Begin RSA Private Key --------" and "----------End of RSA Private Key----------". Don't copy only the alphanumeric key without the header and footer comments. In the Add API Keys dialog, select **Choose Public Key File** to upload your file, or **Paste Public Key**, if you prefer to paste it into a text box and then click **Add**. Copy the fingerprint that you see after you upload the public key in the Console. It looks something like this:`12:34:56:78:90:ab:cd:ef:12:34:56:78:90:ab:cd:ef`.
2. In NetSuite Analytics Warehouse, create the Oracle Object Storage connection using these instructions:
    a. In NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
    b. On the Data Configuration page, click **Manage Connections**.
    c. On the Manage Connections page, click **Create** and then click **Connection**.
    d. In Create Connection, select **Data Extraction** in **Usage Type**, and then select **Oracle Object Storage Service** as the connection type.



e. In the dialog for the Oracle Object Storage Service connection, select **Standard** in **Connectivity Type** and enter these details:

- **Connection Name**: Object Storage
- **Connection Type**: Standard
- **Notification Email**: An email address to receive notifications
- **Remote Host Extract Files Directory**: Name of the base folder in which you must place all your data files in Oracle Object Storage Service
- **URL**: URL of the Oracle Object Storage Service that you noted down in a text file
- **User ID**: OCID of a user that has access to the applicable bucket in Oracle Object Storage Service
- **Finger Print**: The fingerprint that you saw and copied after you uploaded the public key in the Console. It looks something like this: `12:34:56:78:90:ab:cd:ef:12:34:56:78:90:ab:cd:ef`
- **Tenant ID**: Tenancy in the Oracle Infrastructure Cloud Console that you noted down in the text file
- **Private Key**: Paste the private key contents that you previously downloaded
- **File Type**: csv
- **CSV Delimiter**: Delimiter for the data files

- Date format for the data files must match the date format in your Oracle Object Storage Service source; for example, if you've used MM/dd/yyyy (01/23/1998) in your source, then you must specify the same format in **CSV Date Format**. See About Date and Timestamp Formatting for CSV File-based Extractors.
- Timestamp format for the data files must match the timestamp format in your Oracle Object Storage Service source; for example, if you've used MM/dd/yyyy hh:mm:ss (01/23/1998 12:00:20) in your source, then you must specify the same format in **CSV Timestamp Format**



f. Verify that the **Refresh Metadata** toggle is enabled to ensure that the metadata is refreshed when you save the connection.

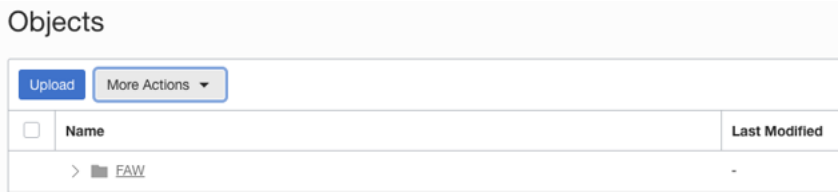
You can later refresh the metadata from the Actions menu on the Manage Connections page, if required.

> **Note:**
>
> You can't create augmentations for the Oracle Object Storage Service unless you perform a metadata extract.

g. Click **Save**.

3. In Oracle Object Storage Service:

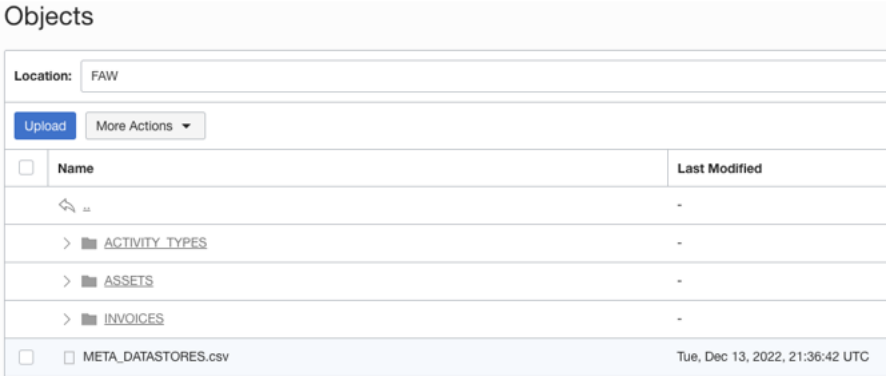
a. Create the folder structure in the Bucket using these guidelines:



- The base folder in the bucket must match with the details provided in the connection.
- Inside the base folder, ensure to place each file in its own folder.
- Ensure that the Prefix of Data_Store_Name (same as Folder name) and Files in the target folder match exactly.

See the "To create a folder or subfolder" section in Using the Console.

b. Inside the base folder, create the metadata file for the Data Store List. This file lists the supported data stores. Each data store is a folder that has the actual file used in data augmentation, for example, ASSETS. Ensure that the file name and folder name match and there aren't any special characters (including space) in the datastore, folder or file names.



c. Create the metadata file for each data file under the data store folder using these guidelines:

The META_DATASTORES.csv must have these columns:

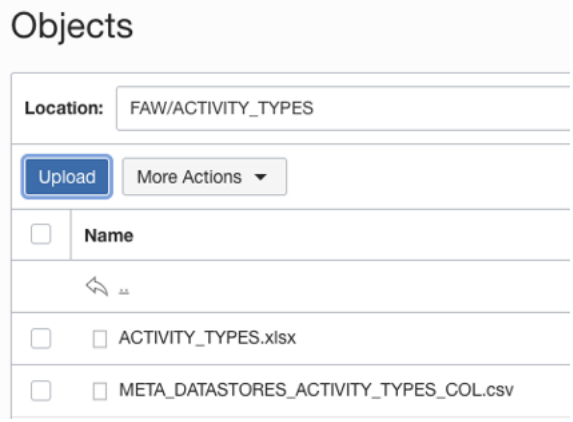
- DATA_STORE_NAME - A mandatory column to identify the data store name.
- DATA_STORE_LABEL - A non-mandatory column that identifies the description of the data store.

Each folder must have:

- A data file that has the actual data that gets loaded into NetSuite Analytics Warehouse. This file must have a prefix with the DATA STORE NAME.
- A metadata file for the list of columns contains all the column information on the data. This file must have a Prefix with META_DATASTORES_<DATA_STORE_NAME>_COL.
  - For the columns in this metadata, ensure the following:
  - If column name is ABC, then metadata can be ABC or “ABC” - the double quotes are ignored.
  - If column name is “ABC”, then metadata must be “”ABC”” – the first double quotes are ignored.

Example

In the image, the folder name is ACTIVITY_TYPES. Hence, the data store name is ACTIVITY_TYPES. You can confirm this from the META_DATASTORES.csv file. In this example, the file is named ACTIVITY_TYPES.xlsx or ACTIVITY_TYPES.csv. The metadata file must be META_DATASTORES_ACTIVITY_TYPES_COL.csv.



The META_DATASTORES_ACTIVITY_TYPES_COL.csv has these columns:

- DATA_STORE_NAME - This is a mandatory column.
- COLUMN_NAME - This is a mandatory column.
- COLUMN_LABEL - This is a non-mandatory column.
- DATA_TYPE – This is a mandatory column.
- WIDTH – This column identifies the string length.
- PRECISION - This column value must be Numeric data type.
- SCALE - This column value must be Numeric data type.
- KEY_SEQUENCE - This is a mandatory column that identifies the Primary Key definition. If you're using the composite primary key, then use column order numbers as values.

4. In NetSuite Analytics Warehouse, on the Manage Connections page, select **Actions** for the Oracle Object Storage Service connection and then select **Test Connection**.
5. After the connections are successfully established, navigate to the Data Configuration page, select the connection in **Data Source**, then click the Data Augmentation tile, and create a data augmentation using the data from the Oracle Object Storage Service. Select the applicable source tables from the Oracle Object Storage Service data. See Augment Your Data.

Load Data from a Secure FTP Source into NetSuite Analytics Warehouse (Preview)

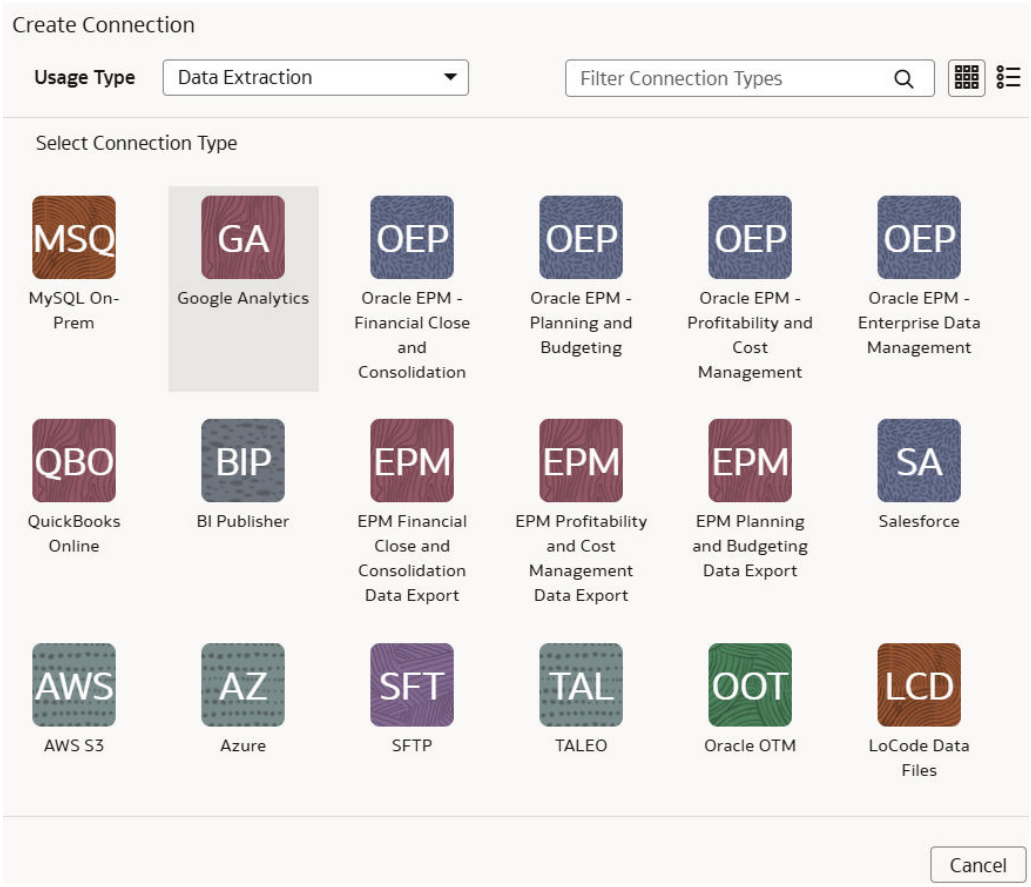
As a service administrator, you can use the NetSuite Analytics Warehouse extract service to acquire data from a secure FTP source (SFTP) and use it to create data augmentations.

Ensure that **SFTP** is enabled on the Enable Features page prior to creating this connection. See Make Preview Features Available.

1. Check the port number for your secure FTP database and create a service request to get the port opened.

> **Note:**
>
> You must provide the IP address of the SFTP server, which should be a public IP and can't be hostname and a fully qualified domain name (FQDN) or a class A private IP.

2. In NetSuite Analytics Warehouse, create the SFTP data connection using these instructions:
   a. In NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
   b. On the Data Configuration page, click **Manage Connections**.
   c. On the Manage Connections page, click **Create** and then click **Connection**.
   d. In Create Connection, select **Data Extraction** in **Usage Type**, and then select **SFTP** as the connection type.

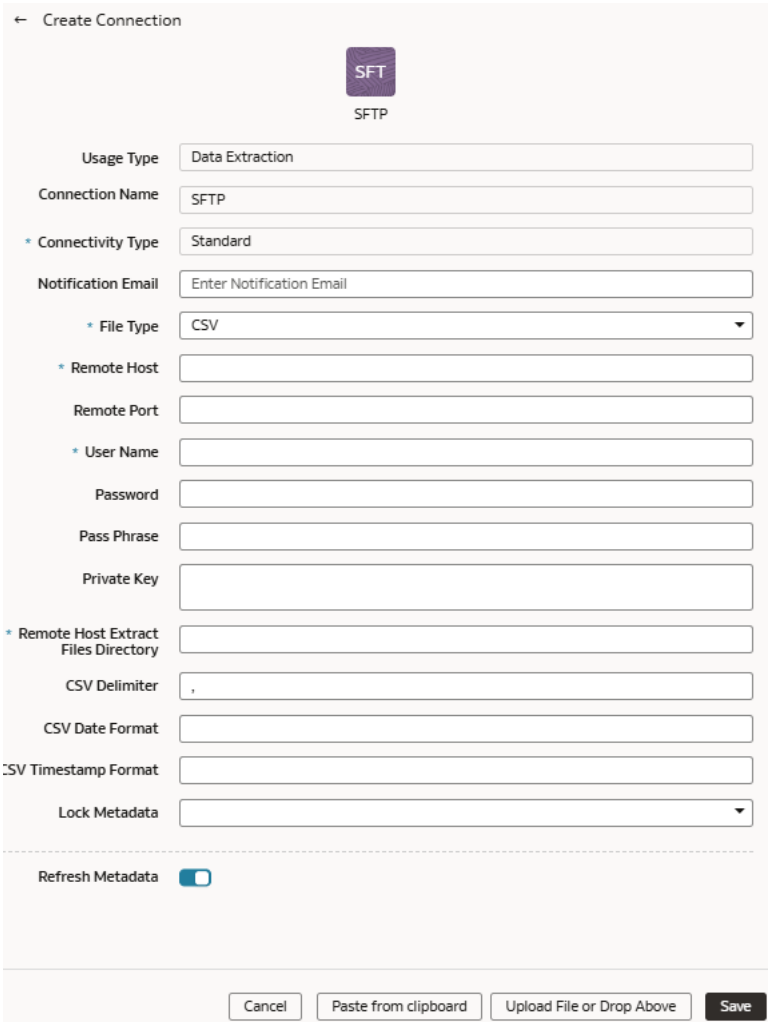


   e. In the dialog for the SFTP connection, select **Standard** in **Connectivity Type**, enter an email address to receive notifications in **Notification Email**, and provide applicable values in **Remote Host**, **User Name**, **Private Key**, **Remote Host Extract Files Directory**, **File Type**, **CSV Delimiter**, **CSV Date Format**, and **CSV Timestamp Format**. In **Lock Metadata**, specify whether you want to turn off the metadata extracts after first refresh if metadata isn't going to change. This option is useful if the flag to derive metadata from data files using the metadata utility is turned on in your source.

In **Remote Host**, ensure that you specify an SFTP Server that supports FIPS Compliant key exchange algorithms.

Ensure the following:

- The table name and file name in your SFTP source needs to be the same.
- The private key you provide is in the valid OpenSSH format and the minimum number of bits in the key should be 2048.
- Date format for the data files must match the date format in your SFTP source; for example, if you've used MM/dd/yyyy (01/23/1998) in your source, then you must specify the same format in **CSV Date Format**. See About Date and Timestamp Formatting for CSV File-based Extractors.
- Timestamp format for the data files must match the timestamp format in your SFTP source; for example, if you've used MM/dd/yyyy hh:mm:ss (01/23/1998 12:00:20) in your source, then you must specify the same format in **CSV Timestamp Format**.



f. Verify that the **Refresh Metadata** toggle is enabled to ensure that the metadata is refreshed when you save the connection.

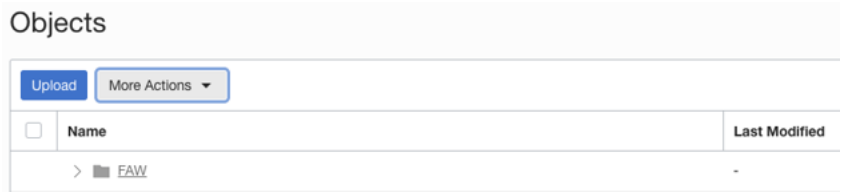
You can later refresh the metadata from the Actions menu on the Manage Connections page, if required.

> **Note:**
>
> You can't create augmentations for SFTP unless you perform a metadata extract.

g. Click **Save**.

3. In your SFTP source:

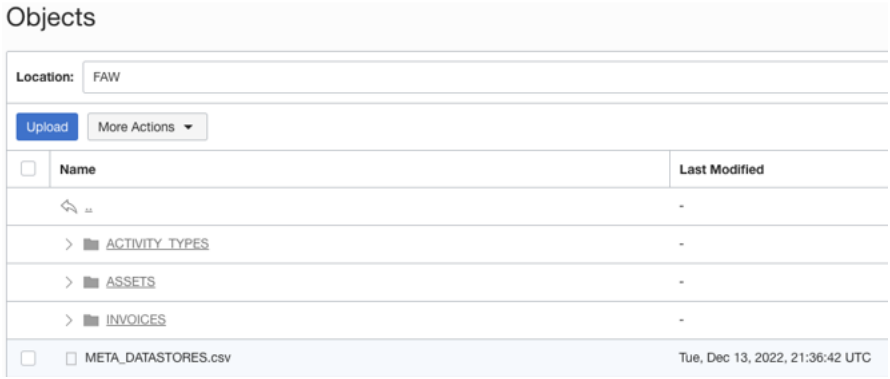
a. Create the folder structure in the Bucket using these guidelines:



- The base folder in the bucket must match with the details provided in the connection.
- Inside the base folder, ensure to place each file in its own folder.
- Ensure that the Prefix of Data_Store_Name (same as Folder name) and Files in the target folder match exactly.

See the "To create a folder or subfolder" section in Using the Console.

b. Inside the base folder, create the metadata file for the Data Store List. This file lists the supported data stores. Each data store is a folder that has the actual file used in data augmentation, for example, ASSETS. Ensure that the file name and folder name match and there aren't any special characters (including space) in the datastore, folder or file names.



c. Create the metadata file for each data file under the data store folder using these guidelines:

The META_DATASTORES.csv must have these columns:

- DATA_STORE_NAME - A mandatory column to identify the data store name.
- DATA_STORE_LABEL - A non-mandatory column that identifies the description of the data store.

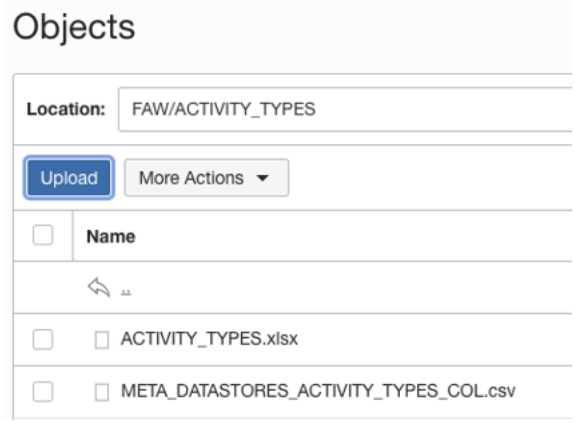
Each folder must have:

- A data file that has the actual data that gets loaded into NetSuite Analytics Warehouse. This file must have a prefix with the DATA STORE NAME.

- A metadata file for the list of columns contains all the column information on the data. This file must have a Prefix with META_DATASTORES_<DATA_STORE_NAME>_COL.
  - For the columns in this metadata, ensure the following:
  - If column name is ABC, then metadata can be ABC or “ABC” - the double quotes are ignored.
  - If column name is “ABC”, then metadata must be ““ABC”” – the first double quotes are ignored.

Example

In the image, the folder name is ACTIVITY_TYPES. Hence, the data store name is ACTIVITY_TYPES. You can confirm this from the META_DATASTORES.csv file. In this example, the file is named ACTIVITY_TYPES.xlsx or ACTIVITY_TYPES.csv. The metadata file must be META_DATASTORES_ACTIVITY_TYPES_COL.csv.



The META_DATASTORES_ACTIVITY_TYPES_COL.csv has these columns:

- DATA_STORE_NAME - This is a mandatory column.
- COLUMN_NAME - This is a mandatory column.
- COLUMN_LABEL - This is a non-mandatory column.
- DATA_TYPE – This is a mandatory column.
- WIDTH – This column identifies the string length.
- PRECISION - This column value must be Numeric data type.
- SCALE - This column value must be Numeric data type.
- KEY_SEQUENCE - This is a mandatory column that identifies the Primary Key definition. If you’re using the composite primary key, then use column order numbers as values.

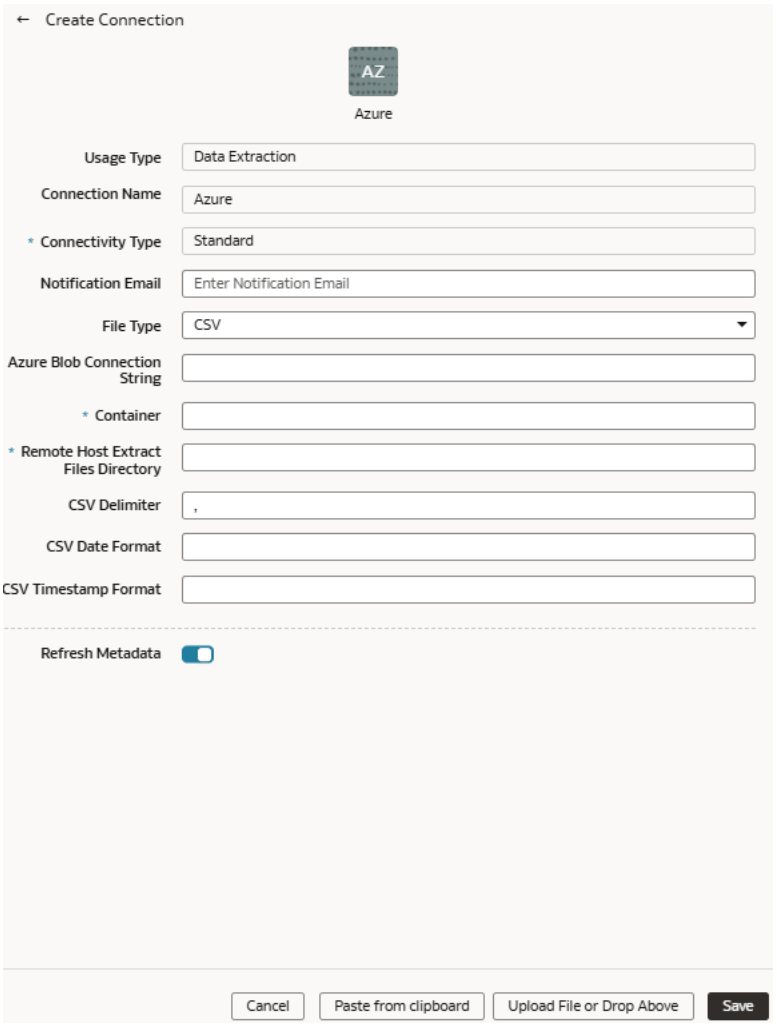
4. On the Manage Connections page, select **Actions** for the SFTP connection and then select **Test Connection**.
5. After the connections are successfully established, navigate to the Data Augmentation tile on the Data Configuration page, select the connection in **Data Source**, and create a data augmentation using the SFTP data. Select the applicable secure FTP source tables. See Augment Your Data.

Load Data from Azure Storage into Oracle NetSuite Analytics Warehouse (Preview)

As a service administrator, you can use the Oracle NetSuite Analytics Warehouse extract service to acquire data from Azure Storage and use it to create data augmentations.

Ensure that **Azure Storage** is enabled on the Enable Features page prior to creating this connection. See Make Preview Features Available.

1. In Oracle NetSuite Analytics Warehouse, create the Azure Storage data connection using these instructions:
    a. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
    b. On the Data Configuration page, click **Manage Connections**.
    c. On the Manage Connections page, click **Create** and then click **Connection**.
    d. In Create Connection, select **Data Extraction** in **Usage Type**, and then select **Azure Storage** as the connection type.
    e. In the dialog for the Azure Storage connection, select **Standard** in **Connectivity Type**, enter an email address to receive notifications in **Notification Email**, enter these Azure Storage instance details, and click **Save**:
        - **Connectivity Type**: Select **Standard**.
        - **Notification Email**: Enter an email address to receive notifications.
        - **File Type**: Select CSV.
        - **Azure Blob Connection String**: Enter your storage account's connection string.
        - **Container**: Specify the Azure container.
        - **Remote Host Extract Files Directory**: Name of the base folder in which you must place all your data files in Azure Storage.
        - **CSV Delimiter**: Delimiter for the data files.
        - **CSV Date Format**: Date format for the data files must match the date format in your Azure Storage source; for example, if you've used MM/dd/yyyy (01/23/1998) in your source, then you must specify the same format. See About Date and Timestamp Formatting for CSV File-based Extractors



f. Verify that the **Refresh Metadata** toggle is enabled to ensure that the metadata is refreshed when you save the connection.

   You can later refresh the metadata from the Actions menu on the Manage Connections page, if required.

> **Note:**
>
> You can't create augmentations for Azure Storage unless you perform a metadata extract.

g. Click **Save**.

2. On the Manage Connections page, select **Actions** for the Azure Storage connection and then select **Test Connection**.
3. After the connections are successfully established, navigate to the Data Configuration page, select the connection in **Data Source**, then click the Data Augmentation tile, and create a data augmentation using the Azure Storage data. Select the applicable Azure Storage source tables. See Augment Your Data.

Connect With Cloud Sources

Connect with your cloud applications to provide the background information for reports.

You can blend the additional data from these data sources with the prebuilt datasets to enhance business analysis. To know about the date and timestamp formatting for the CSV file-based extractors, see About Date and Timestamp Formatting for CSV File-based Extractors.

Create Additional NetSuite Connections

You can create additional connections to the NetSuite source based on your customer tier. These additional connections enable you to bring in data from multiple NetSuite accounts.

You may want to connect to additional NetSuite accounts if you've:

- One NetSuite primary account with live data and other NetSuite accounts having static data.
- More than one NetSuite account with live data but no data mash-up required.
- More than one NetSuite account with live data requiring data mash-up.

As a premium license tier Oracle NetSuite Analytics Warehouse customer, you can connect to two additional NetSuite accounts. If you're an enterprise license tier customer, then you can connect to ten additional NetSuite accounts. Based on your license tier, Oracle displays the additional NetSuite connections on the Create Connection dialog.

1. In NetSuite, complete these tasks:
   - Enable the Multi-Instance Connector (MIC) feature.
   - Add an integration record.
   - Create a token-based authentication permission and add it to a user.
   - Create a token.

   See Merging Your NetSuite Accounts Data Into Your NetSuite Analytics Warehouse Instance
2. In Oracle NetSuite Analytics Warehouse, **Console**, click **Data Configuration** under **Application Administration**.
3. On the Data Configuration page, click **Manage Connections**.
4. On the Manage Connections page, click **Create** and then click **Connection**.
5. In Create Connection, select **Data Extraction** in **Usage Type** and then select **NetSuite 2** or an applicable NetSuite option such as NetSuite 3 or NetSuite 4 as the connection type.
6. In **Create Connection**, enter these details and then click **Save**:
   - **Notification Email**: Enter an email address to receive notifications.
   - **User Name**: Enter your user name.
   - **JDBC URL**: Enter the following URL: jdbc:ns://<*NS Account ID*>.connect.api.NetSuite.com:1708;ServerDataSource=NetSuiteDW;Encrypted=1;CustomProperties=(AccountID=<*NS Account ID*>;RoleID=57;Uppercase=1). Replace <*NS Account ID*> with your NetSuite Account ID.
   - **Account ID**: Enter your NetSuite Account ID.
   - **Consumer Key**: Paste the Consumer Key / Client ID string you noted down after creating the integration record.

- **Consumer Secret**: Paste the Consumer Secret / Client Secret string you noted down after creating the integration record.
- **Token Key**: Paste the Token ID string you noted down after creating the token.
- **Token Secret**: Paste the Token Secret string you noted down after creating the token.

7. Verify that the **Refresh Metadata** toggle is enabled to ensure that the metadata is refreshed when you save the connection.

   You can later refresh the metadata from the Actions menu on the Manage Connections page, if required.

   > **Note:**
   >
   > You can't create augmentations for NetSuite 2 unless you perform a metadata extract.

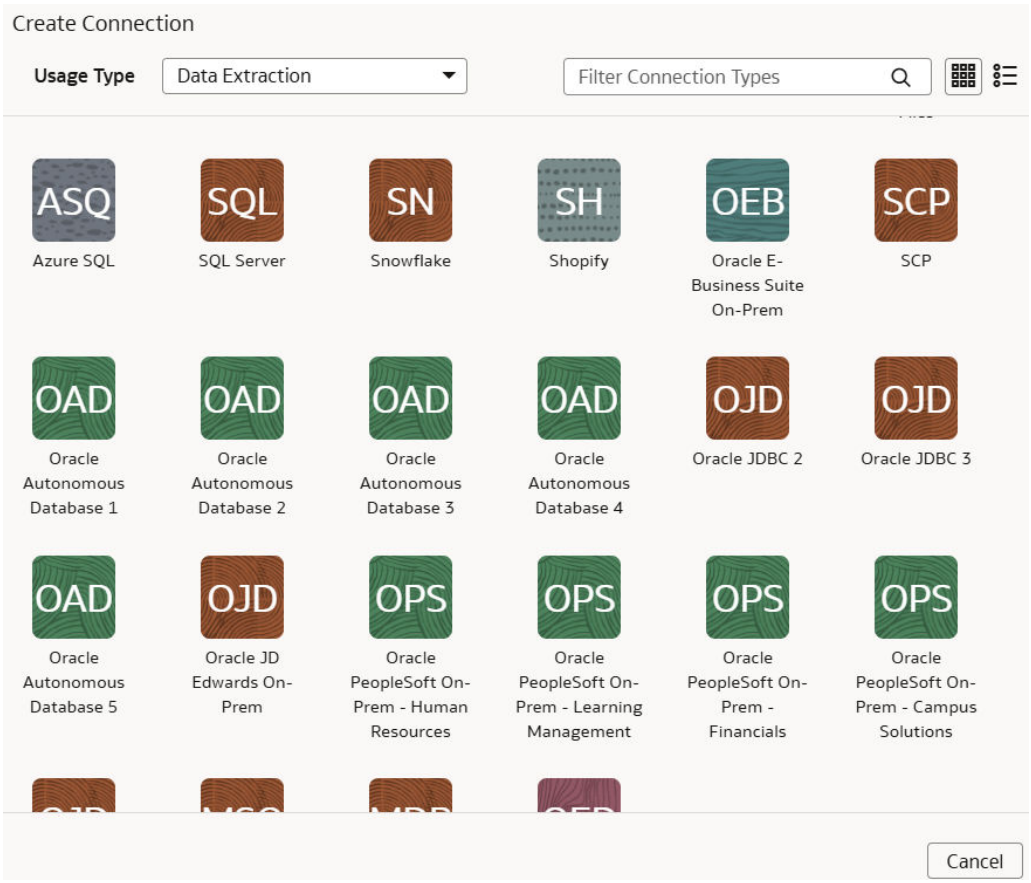
8. On the Manage Connections page, select **Actions** for the NetSuite 2 connection and then select **Test Connection**.
9. After the connections are successfully established, navigate to the Data Configuration page, select the connection in **Data Source**, then click the Data Augmentation tile, and create a data augmentation using the NetSuite 2 data. Select the applicable NetSuite 2 source tables. See Augment Your Data.

Load Data from Azure SQL into Oracle NetSuite Analytics Warehouse (Preview)

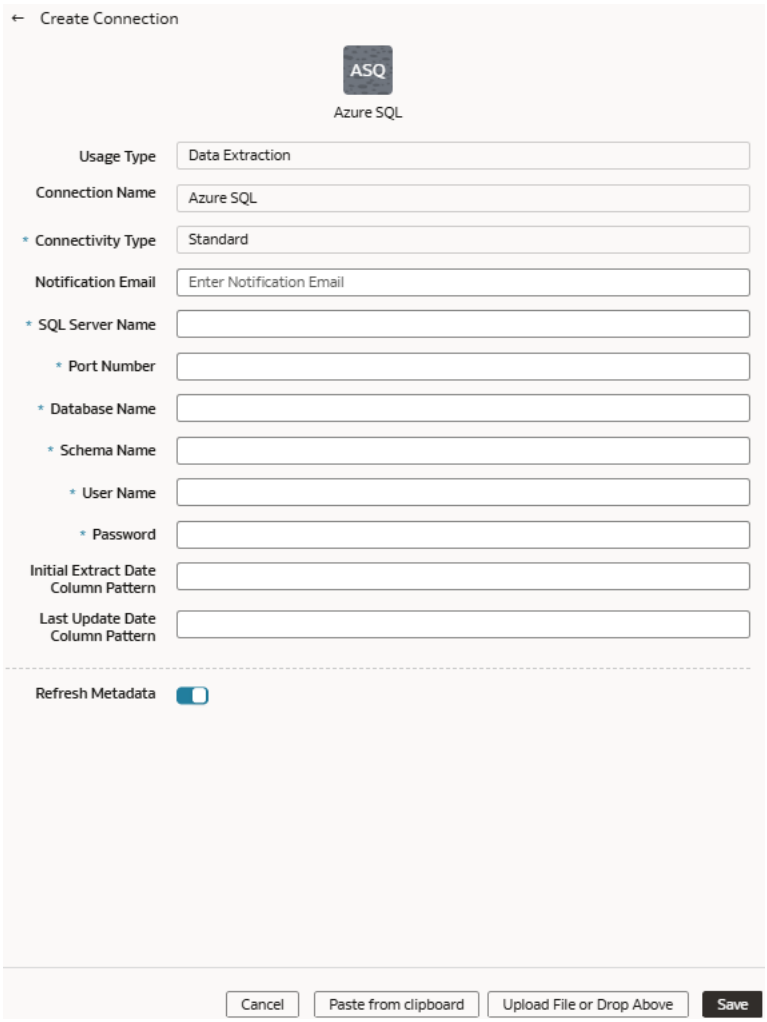
As a service administrator, you can use the Oracle NetSuite Analytics Warehouse extract service to acquire data from Azure SQL and use it to create data augmentations.

Ensure that **Azure SQL** is enabled on the Enable Features page prior to creating this connection. See Make Preview Features Available.

1. In Oracle NetSuite Analytics Warehouse, create the Azure SQL data connection using these instructions:
   a. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
   b. On the Data Configuration page, click **Manage Connections**.
   c. On the Manage Connections page, click **Create** and then click **Connection**.
   d. In Create Connection, select **Data Extraction** in **Usage Type** and then select **Azure SQL** as the connection type.



e. In the dialog for the Azure SQL connection, select **Standard** in **Connectivity Type**, enter an email address to receive notifications in **Notification Email**, enter your Azure SQL instance details, and click **Save**.



f. Verify that the **Refresh Metadata** toggle is enabled to ensure that the metadata is refreshed when you save the connection.

   You can later refresh the metadata from the Actions menu on the Manage Connections page, if required.

> **Note:**
>
> You can't create augmentations for Azure SQL unless you perform a metadata extract.

2. On the Manage Connections page, select **Actions** for the Azure SQL connection and then select **Test Connection**.
3. After the connections are successfully established, navigate to the Data Configuration page, select the connection in **Data Source**, then click the Data Augmentation tile, and create a data augmentation using the Azure SQL data. Select the applicable Azure SQL source tables. See Augment Your Data.

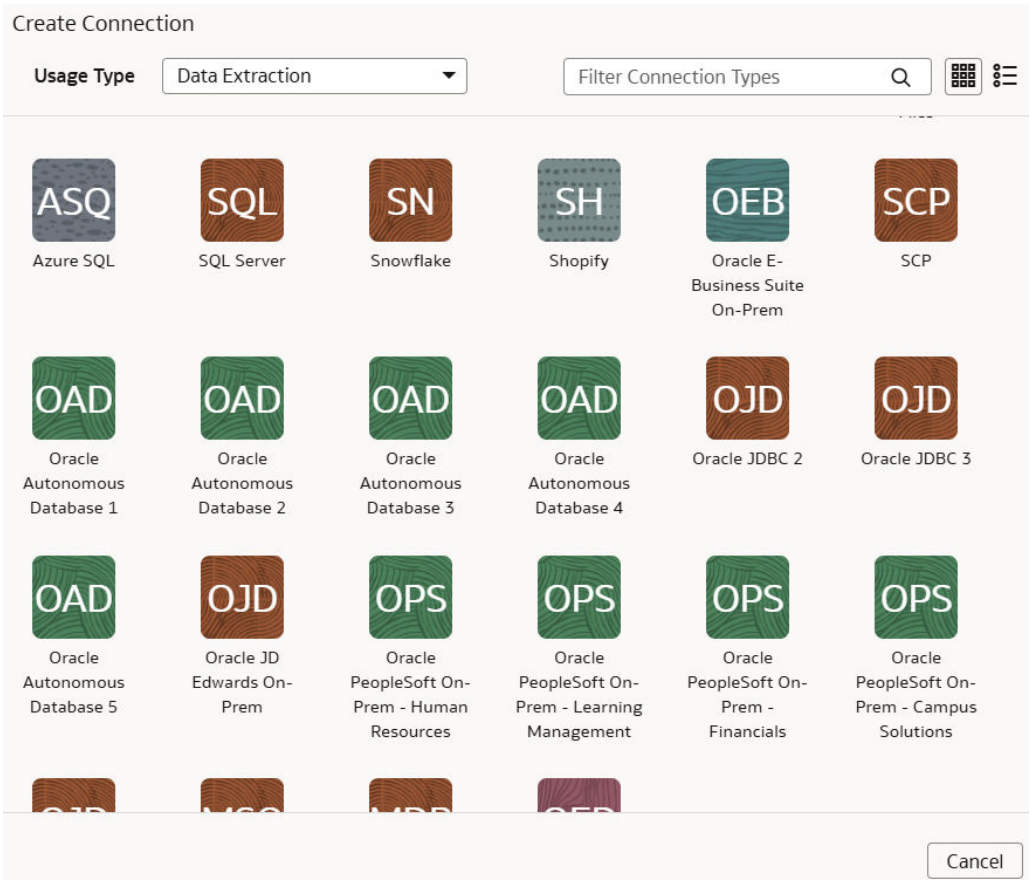
Load Data from Oracle Autonomous Database into Oracle NetSuite Analytics Warehouse (Preview)

As a service administrator, you can use the Oracle NetSuite Analytics Warehouse extract service to acquire data from Oracle Autonomous Database and use it to create data augmentations.

You can create connections to five autonomous databases. Depending on the number of connections, ensure that options such as **Oracle Autonomous Database 1**, **Oracle Autonomous Database2** are enabled on the Enable Features page prior to creating this connection. See Make Preview Features Available.

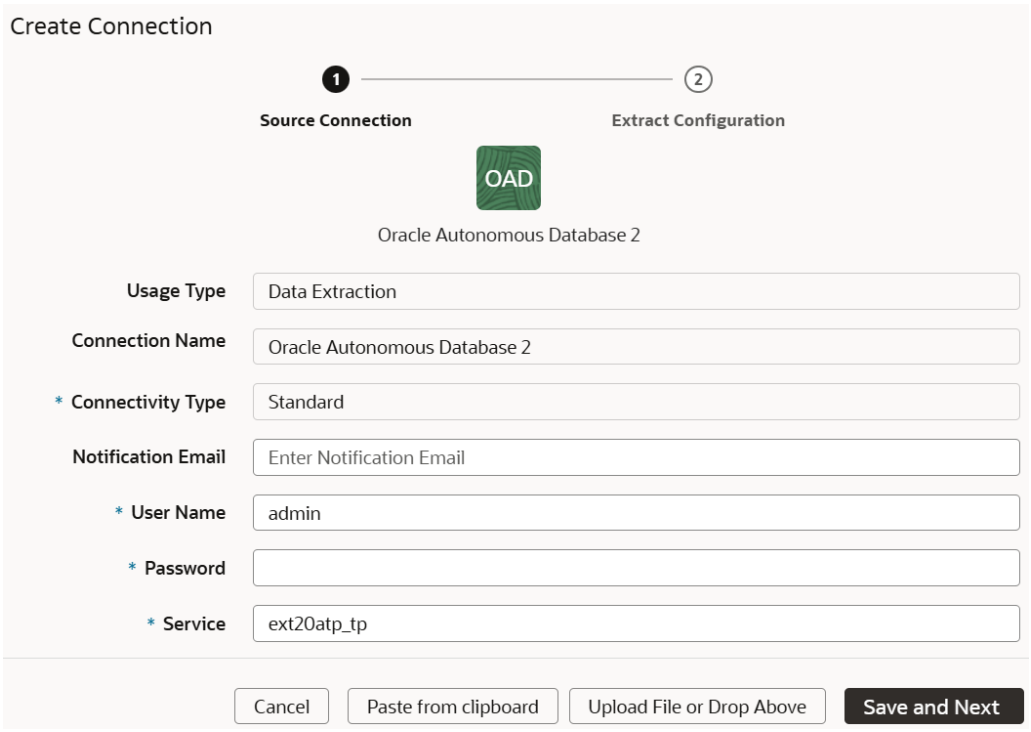
> **Note:**
>
> Currently, you can't connect to a private autonomous transaction processing database (ATP database).

1. In Oracle NetSuite Analytics Warehouse, create the autonomous database connection using these instructions:
   a. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
   b. On the Data Configuration page, click **Manage Connections**.
   c. On the Manage Connections page, click **Create** and then click **Connection**.
   d. In Create Connection, select **Data Extraction** in **Usage Type** and depending on the number of connections, select options such as **Oracle Autonomous Database 1**, or **Oracle Autonomous Database2** as the connection type.



e. In the dialog for the Oracle Autonomous Database connection, provide these details in Source Connection and then click **Save and Next**:

- Select **Standard** in **Connectivity Type**.
- Enter an email address to receive notifications in **Notification Email**.
- Enter the credentials to access the database in **User Name** and **Password**.
- Enter the database service details in **Service**.
- In **Wallet**, drag and drop the database wallet details.



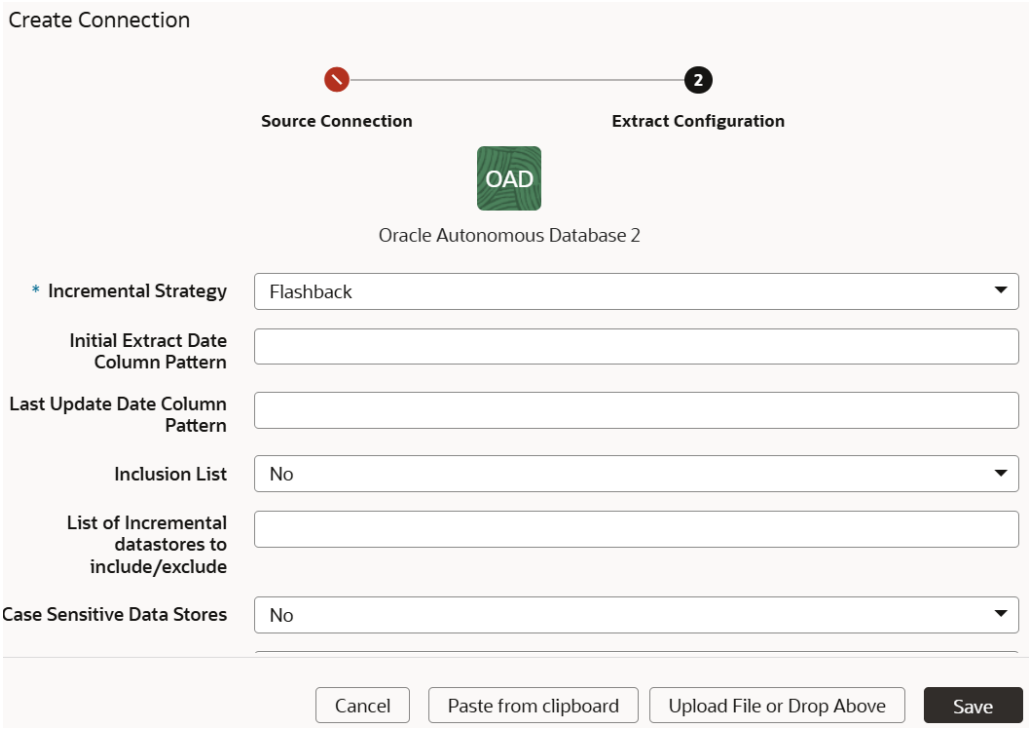
f. In the dialog for the Oracle Autonomous Database connection, provide these details in Extract Configuration and click **Save**:

- In **Incremental Strategy**, select the incremental strategy (Flashback, Golden Gate, or RowSCN) that is compatible with your database configuration.
- In **Initial Extract Date Column Pattern**, provide the initial extract date pattern that matches the pattern in your source.
- In **Last Update Date Column Pattern**, provide the last update date pattern that matches the pattern in your source.
- In **Inclusion List**, select **Yes** or **No** to include the mentioned list of datastores in the incremental strategy or not.
- In **List of Incremental datastores to include/exclude**, enter a comma separated list of datastores names.

  If you don't provide, then the connector uses the incremental strategy for all the datastores. If you provide and IS_INCLUSION_LIST=true, only the provided list use the specified incremental strategy. If provided and IS_INCLUSION_LIST=false, the provided list won't use the incremental strategy. If INCREMENTAL_STRATEGY property is available, then the connector uses IS_INCLUSION_LIST and INCR_DATASTORES_LIST for all strategies. If not available, then for FLASHBACK the connector checks if ENABLE_FLASHBACK_INCREMENTAL and FBA_INCR_DATASTORES_EXCEPTION_LIST is provided and for ROWSCN, it checks if ENABLE_ORA_ROWSCN_INCREMENTAL and ROWSCN_INCR_DATASTORES_EXCEPTION_LIST is provided.
- In **Case sensitive Data Stores**, select **Yes** or **No**.
- In **Schema Name**, enter the schema name to extract data from.

- In **Data Store Name Pattern**, specify the name pattern of the data stores that you want extracted. If you provide this value, then the connector extracts only data stores matching the pattern.
- Verify that the **Refresh Metadata** toggle is enabled to ensure that the metadata is refreshed when you save the connection. You can later refresh the metadata from the Actions menu on the Manage Connections page, if required.

> **Note:**
>
> You can't create augmentations for autonomous database unless you perform a metadata extract.



2. On the Manage Connections page, select **Actions** for the autonomous database connection and then select **Test Connection**.
3. After the connections are successfully established, navigate to the Data Configuration page, select the connection in **Data Source**, then click the Data Augmentation tile, and create a data augmentation using the autonomous database data. Select the applicable autonomous database source tables. See Augment Your Data.

Load Data from Enterprise Data Management Cloud into NetSuite Analytics Warehouse (Preview)

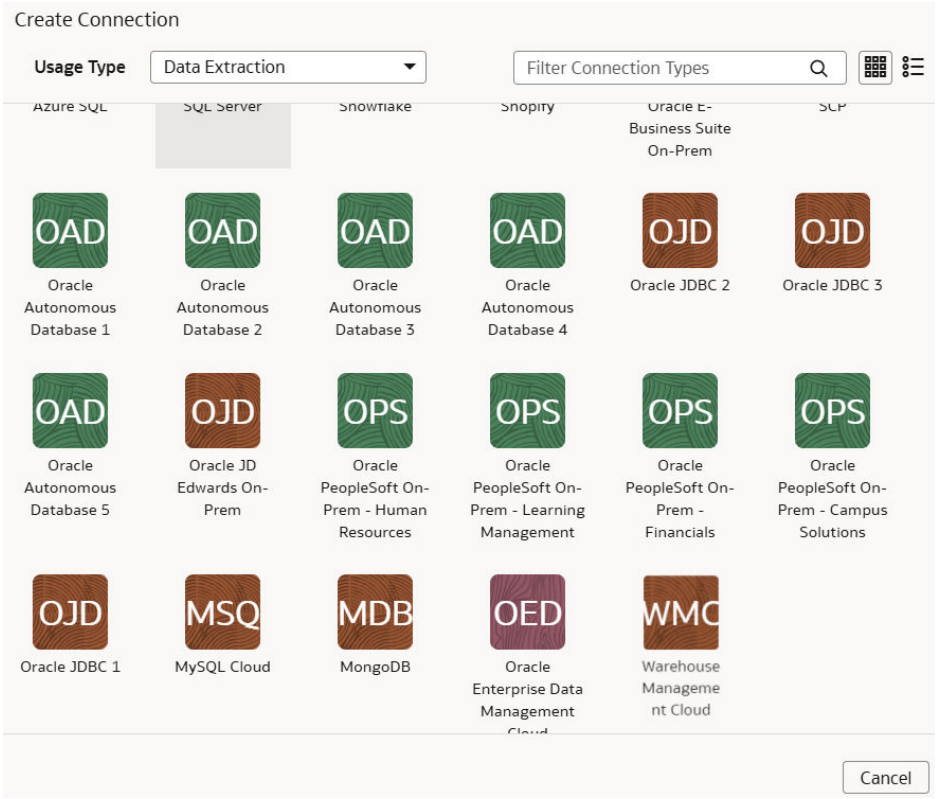
As a service administrator, you can use the NetSuite Analytics Warehouse extract service to acquire data from the Enterprise Data Management Cloud instance and use it to create data augmentations.

The extracts created in the Enterprise Data Management Cloud service need to be public, hence you must promote your private extracts to the public. Review the documentation and

error messages for the metadata refresh failures for the private extract. This connector supports only the CSV data format.

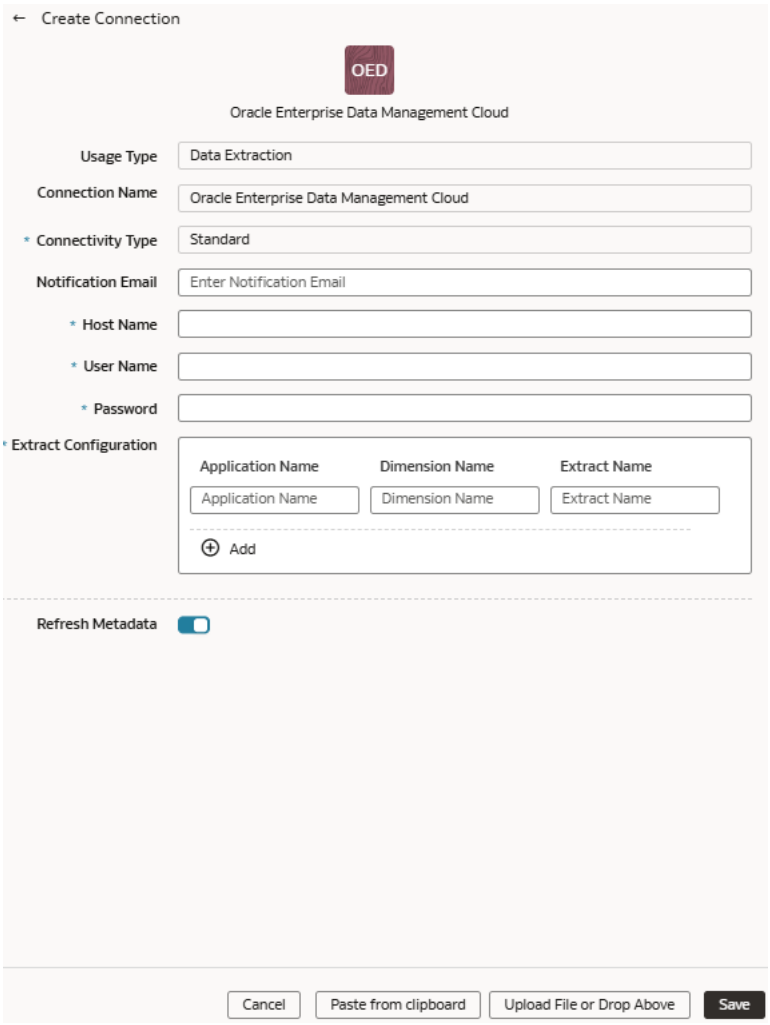
Ensure that **Oracle Enterprise Data Management Cloud** is enabled on the Enable Features page prior to creating this connection. See Make Preview Features Available.

1. In NetSuite Analytics Warehouse, create the Enterprise Data Management Cloud data connection using these instructions:
   a. In NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
   b. On the Data Configuration page, click **Manage Connections**.
   c. On the Manage Connections page, click **Create** and then click **Connection**.
   d. In Create Connection, select **Data Extraction** in **Usage Type** and then select **Oracle Enterprise Data Management Cloud** as the connection type.



   e. In the dialog for the Enterprise Data Management Cloud connection, select **Standard** in **Connectivity Type**, enter an email address to receive notifications in **Notification Email**, host name of the Oracle Enterprise Data Management server in **Host Name**, credentials to access the Enterprise Data Management Cloud instance in **User Name** and **Password**. In **Extract Configuration**, enter the list of extracts using only “comma” as the delimiter. The configuration extract must be a single line JSON without formatting for the quotation marks (" instead of \"), for example:

```
[{"applicationName": "Account Reconciliation", "dimensionName":
"Profiles","extractName": "Profiles"}]
```



f. Verify that the **Refresh Metadata** toggle is enabled to ensure that the metadata is refreshed when you save the connection.

   You can later refresh the metadata from the Actions menu on the Manage Connections page, if required.

> **Note:**
>
> You can't create augmentations for Enterprise Data Management Cloud unless you perform a metadata extract.

g. Click **Save**.

2. On the Manage Connections page, select **Actions** for the Enterprise Data Management Cloud connection and then select **Test Connection**.

3. After the connections are successfully established, navigate to the Data Configuration page, select the connection in **Data Source**, then click the Data Augmentation tile, and create a data augmentation using the Enterprise Data Management Cloud data. Select the applicable Enterprise Data Management Cloud source tables. See Augment Your Data.

Load Data from Enterprise Performance Management into NetSuite Analytics Warehouse (Preview)

As a service administrator, you can use the NetSuite Analytics Warehouse extract service to acquire data from the Enterprise Performance Management (EPM) SaaS instance and use it to create data augmentations for various Enterprise Resource Planning and Supply Chain Management use cases.

You can connect to these functional modules of EPM:

- Financial Close and Consolidation (FCCS)
- Planning and Budgeting (PBCS)
- Profitability and Cost Management (PCMCS)

If you've pivot table in your source, then the metadata extract supports pivot table metadata extraction for the EPM connectors. You can only manually extract the incremental data because, for incremental extraction, you must update the results file in EPM before starting the next extraction for the updated data. Update the results file by running the integration using Data Exchange and then access the new results file from the EPM connector in NetSuite Analytics Warehouse. Ensure that you enter all the fields in accordance with your EPM nomenclature and specifics. For example, if you have defined Period in your EPM job as {June-23}, then include exactly the same in the Create Connection for the EPM source dialog.

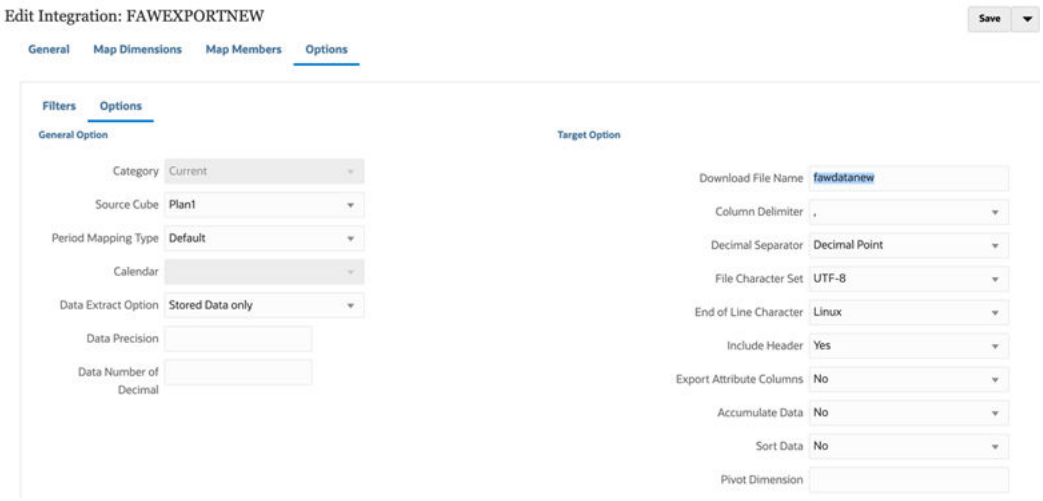
> **Note:**
>
> The EPM connectors display the default datatype and size; you must edit these values as applicable while creating data augmentations.

Depending on the functional module you want to connect to, ensure that the applicable feature is enabled on the Enable Features page prior to creating this connection:

- **Oracle EPM - Financial Close and Consolidation**
- **Oracle EPM - Planning and Budgeting**
- **Oracle EPM - Profitability and Cost Management**

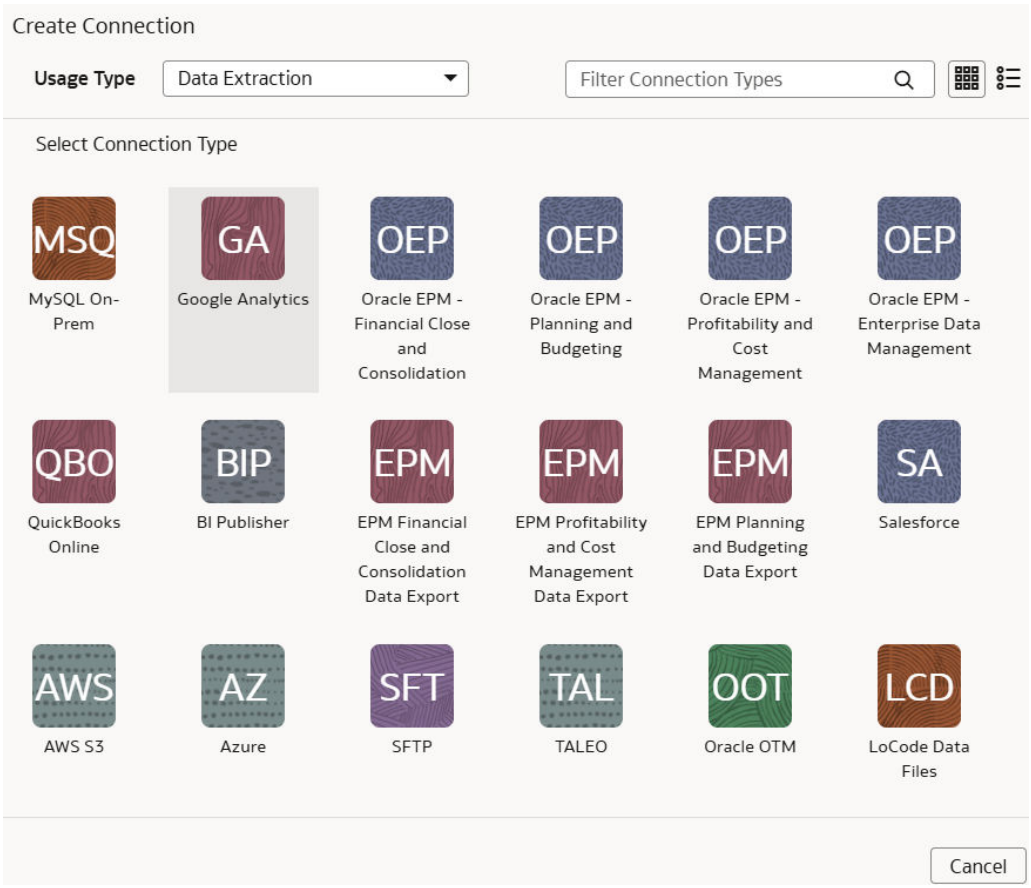
See Make Preview Features Available.

1. In EPM, create an integration, write out the results into a file whose name you provide in **Download File Name**, and then specify that same file name in List of Data Files while creating the connection to EPM in NetSuite Analytics Warehouse to extract the data.



2. In EPM, when exporting data, use one of the following modes:
   - Standard mode: This built-in workflow process helps manage the data flow process in and out of EPM. For Standard mode, you specify the period when you run the integration.
   - Quick mode: This process by-passes most of the steps in the workflow and provides a significant improvement in the extract process. When using quick mode, you specify the period in the integration definition filter, and this may include a substitution variable.When using the API, the same rules apply, except that when you need to specify a period, you can't use a substitution variable to pull from EssBase, but will need to include the period or year on extract.

   See Exporting Data.
3. In NetSuite Analytics Warehouse, create the EPM data connection using these instructions:
   a. In NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
   b. On the Data Configuration page, click **Manage Connections**.
   c. On the Manage Connections page, click **Create** and then click **Connection**.
   d. In Create Connection, select **Data Extraction** in **Usage Type** and then select the connection type based on the functional module that you want to connect to. For example, to connect to the "Financial Close and Consolidation (FCCS)" module, select **Oracle EPM - Financial Close and Consolidation** as the connection type.



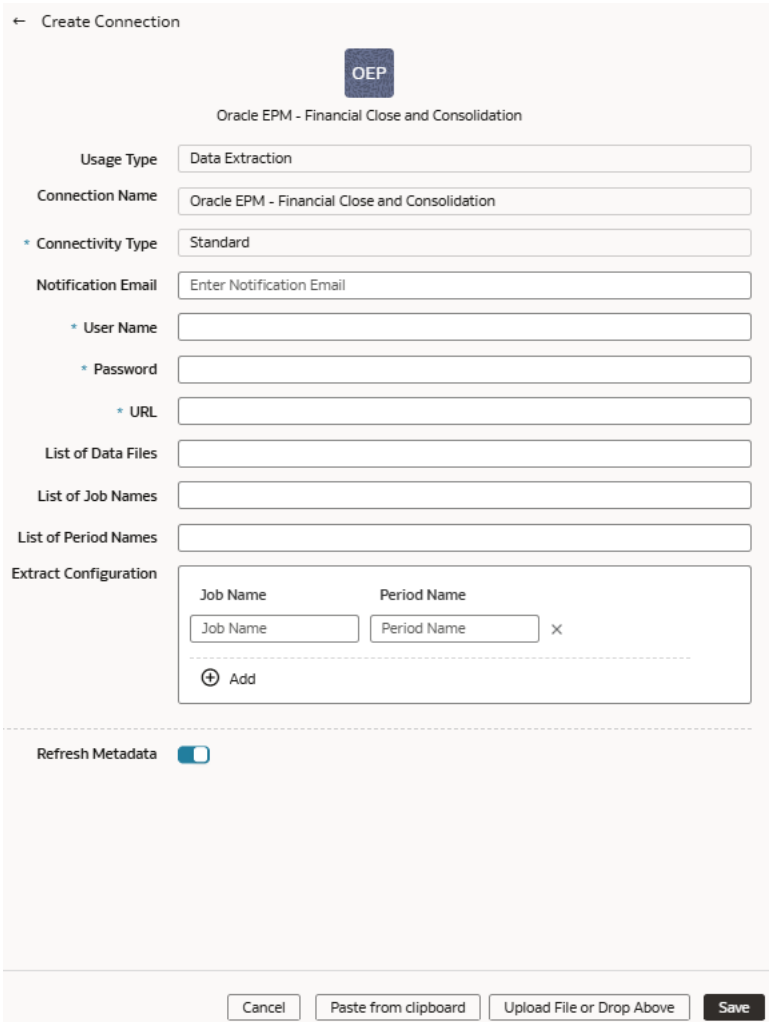
e. In Create Connection for the EPM source, enter these details and click **Save**:

- **Connectivity Type**: Select **Standard**.
- **Notification Email**: Enter an email address to receive notifications.
- **User Name** and **Password**: Enter the credentials for your EPM source. Prefix the user name with the domain of your EPM source, such as `domain.username`.
- **URL**: Enter the specific URL of your EPM source using the `https://<DOMAIN_NAME>region.ocs.oraclecloud.com` format. For example, `https://epm7-test-a123456.epm.us6.oraclecloud.com`. Ensure to use the `https://` protocol to avoid a timeout error.
- **List of Data Files**: Specify the file name that you had entered in **Download File Name** while creating an integration in EPM.
- **List of Job Names**: To ensure a healthy connection, you should first run the desired EPM integration jobs in the EPM user interface. Then enter the successfully-run and valid EPM integration job names in this field. If you've multiple job names, ensure that they are comma separated. For example, `FAWEPMTestingV2, FAWEXPORT,FAW Job Testing`.
- **List of Period Names**: Provide the period names for the corresponding job names. Ensure that multiple period names are comma separated. You may leave this blank, in which case the connector uses the global point of view. Few examples of period names are:
  - `{Jan-22},{Oct-22}`
  - `{Nov-22}{Dec-22},{Jan-23}{Feb-23}`

  - `{Jan#FY20}{Mar#FY20},{Apr#FY20}{May#FY20}`

  For elaborate examples, refer to Running Integrations.

- **Extract Configuration**: Provide the EPM job name such as "Job 1" and period name such as "Quarter 1" corresponding to the given job name to extract data for specific periods. This enables you to run multiple EPM jobs.



f. Verify that the **Refresh Metadata** toggle is enabled to ensure that the metadata is refreshed when you save the connection.

   You can later refresh the metadata from the Actions menu on the Manage Connections page, if required.

   > **Note:**
   >
   > You can't create augmentations for EPM unless you perform a metadata extract.

g. Click **Save**.

4. On the Manage Connections page, select **Actions** for the EPM connection and then select **Test Connection**.

5. After the connections are successfully established, navigate to the Data Configuration page, select the connection in **Data Source**, then click the Data Augmentation tile, and create a data augmentation using the EPM data. Select the applicable EPM source tables. See Augment Your Data.

Load Data from Google Analytics into NetSuite Analytics Warehouse

As a service administrator, you can use the NetSuite Analytics Warehouse extract service to acquire data from the Google Analytics SaaS instance and use it to create data augmentations for various Enterprise Resource Planning and Supply Chain Management use cases.

Before connecting with the Google Analytics source, note these:

- NetSuite Analytics Warehouse supports Google Analytics extractor for GA4 properties and doesn't support the previous version – Google Universal Analytics (UA) properties.
- DataStores are the list of GA4 properties.
- DataStore columns are the list of Dimensions and Metrics for a GA4 property.
- DataExtract runs the report based on user selection for a GA4 property as DataStore and Dimensions and Metrics as DataStore columns.
- MetaExtract fetches metadata for all the available GA4 properties (DataStores) and its Dimensions and Metrics (DataStoreColumns).
- This connector supports limited number of Google Analytics metrics. See Dimensions Metrics Explorer to know what is available.

1. In Google Cloud (Analytics) Project, create a service account and download the `credentials.json` file.

   See https://cloud.google.com/docs/authentication/production.

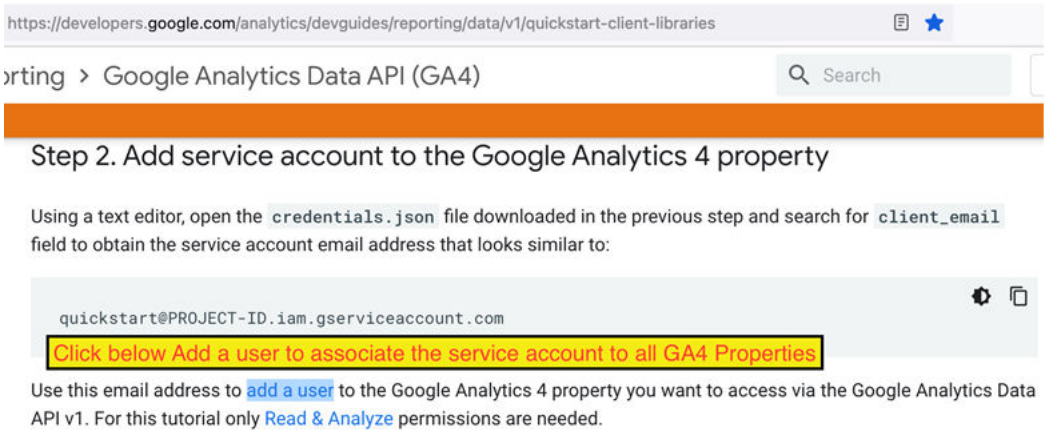
2. Add the service account to the Google Analytics 4 property.

   See https://developers.google.com/analytics/devguides/reporting/data/v1/quickstart-client-libraries.

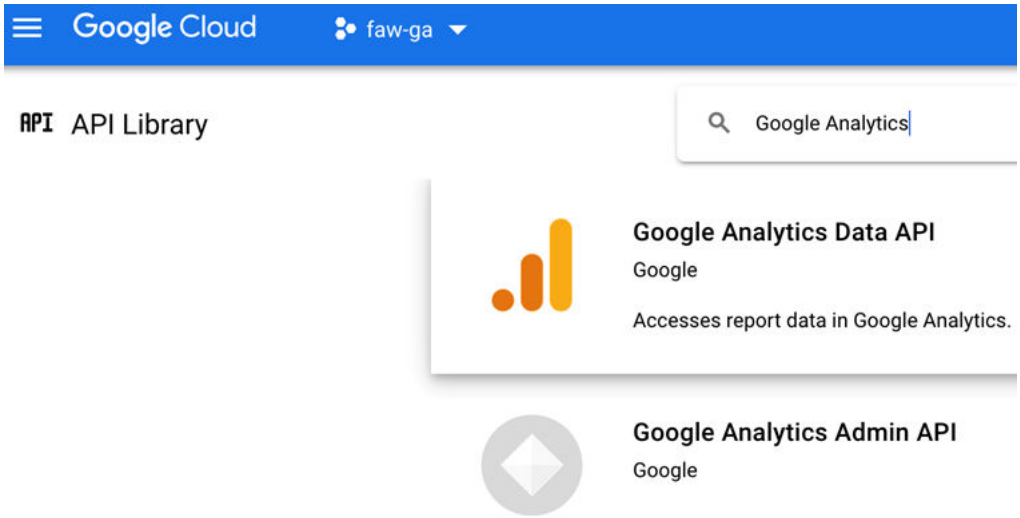
3. Enable Google Analytics APIs using these instructions:

   a. Using a text editor, open the `credentials.json` file that you had downloaded and search for the `client_email` field to obtain the service account email address.

   b. Use this email address to add a user to the Google Analytics 4 property you want to access through the Google Analytics Data API v1.



4. Ensure that the Google Analytics Admin API, Google Analytics Data API are available for your Google Analytics instance.



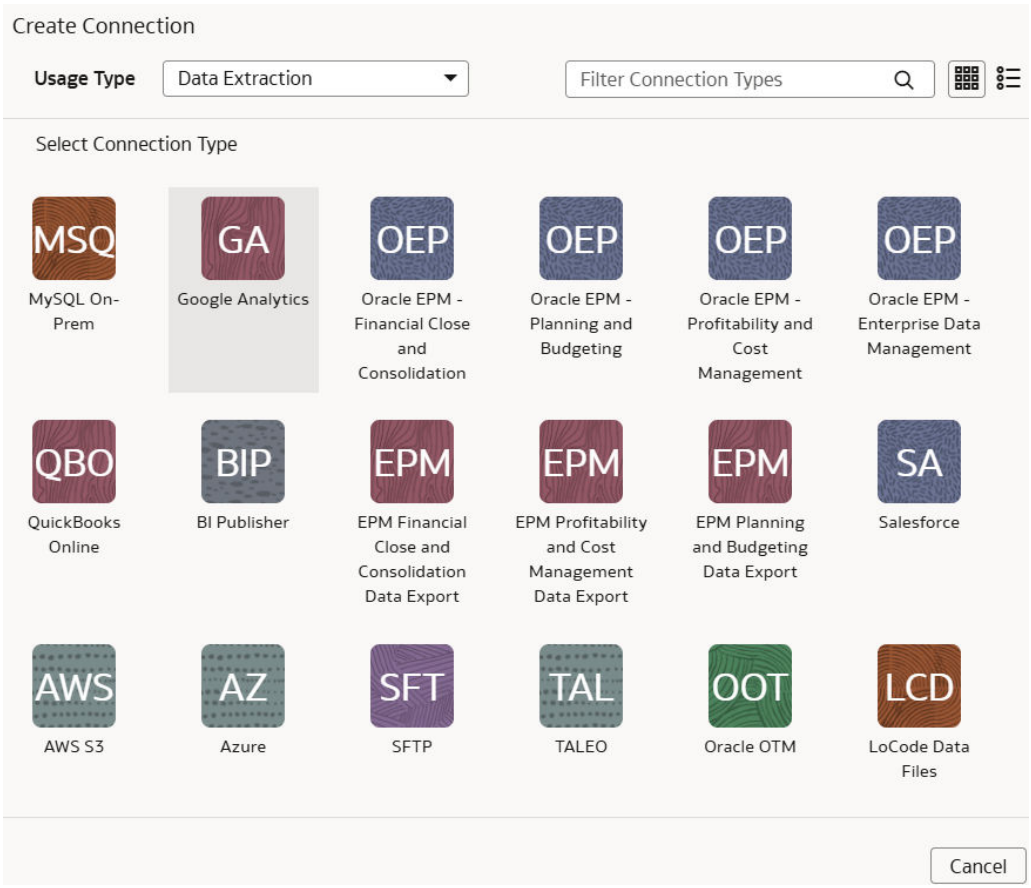
5. In NetSuite Analytics Warehouse, create the Google Analytics data connection using these instructions:

   a. In NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.

   b. On the Data Configuration page, click **Manage Connections**.

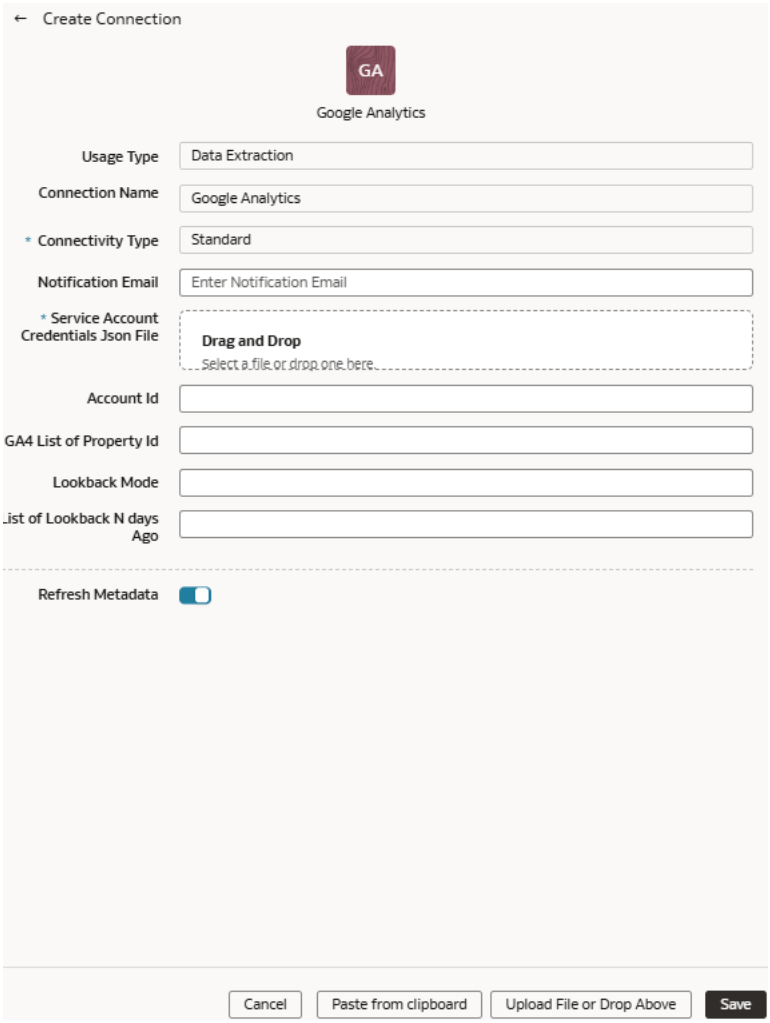
   c. On the Manage Connections page, click **Create** and then click **Connection**.

   d. In Create Connection, select **Data Extraction** in **Usage Type** and then select **Google Analytics** as the connection type.



e. In the dialog for the Google Analytics connection, select **Standard** as the connectivity type and enter these details:

- **Notification Email**: An email address to receive notifications regarding this connection.
- **Service Account Credentials Json File**: The Google Cloud Service Account `credentials.json` file that you had downloaded.
- **Account ID**: Google Analytics account ID.
- **GA4 List of Property ID**: The GA4 List of Property ID with commas to separate each ID.
- **Lookback Mode**: Select either Full or Committed.
- **List of Lookback N days Ago**: Comma separated list of days (integer) values such as 7,21.



Note these:

- For the Lookback mode, if you don't provide a value, then the Lookback mode isn't supported. The Full option requires one day value, if you provide multiple values, then the process uses the first value. You can provide multiple values for the Committed option.

- For List Data Stores, the REST API returns a list of GA4 Property IDs either using the Account ID (if provided) or just the source configured or provided list of property.
- For List columns, the REST API returns a list of column metadata for the given GA4 Property ID.

f. Verify that the **Refresh Metadata** toggle is enabled to ensure that the metadata is refreshed when you save the connection.

You can later refresh the metadata from the Actions menu on the Manage Connections page, if required.

> **Note:**
>
> You can't create augmentations for Google Analytics unless you perform a metadata extract.
> Metadata extract:
>
> - Retrieves metadata columns for each GA4 Property ID provided in the source configuration.
> - Prefixes the GA property columns with `Dimension_` or`Metric_` that NetSuite Analytics Warehouse later uses while extracting data to differentiate Dimension and Metric column type.
> - Leaves the payload `dataStores` array empty.

g. Click **Save**.

6. On the Manage Connections page, select **Actions** for the Google Analytics connection and then select **Test Connection**.

> **Note:**
>
> REST API signature is same across sources. Test connection invokes GA Common Metadata API. This returns the default version values and no calls are made to the source.

7. After the connections are successfully established, navigate to the Data Configuration page, select the connection in **Data Source**, then click the Data Augmentation tile, and create a data augmentation using the Google Analytics data. Select the applicable Google Analytics source tables. Ensure that you specify "Dimension_transactionId" in the source tables as the primary key and use it to join each of the data augmentation tables. You can select a maximum number of nine dimensions for each data augmentation. See Augment Your Data.

   When you enable data extraction, you can schedule to run when you choose to do so. For data extraction, note these:

   a. Provide date ranges to run the report and fetch data.

   b. Regular data extract uses the initial or last ExtractDate as *StartDate* and job RunDate as *EndDate*.

   c. Lookback mode includes additional date ranges along with the regular extract date range which fetches additional data set but in a single runReport call.

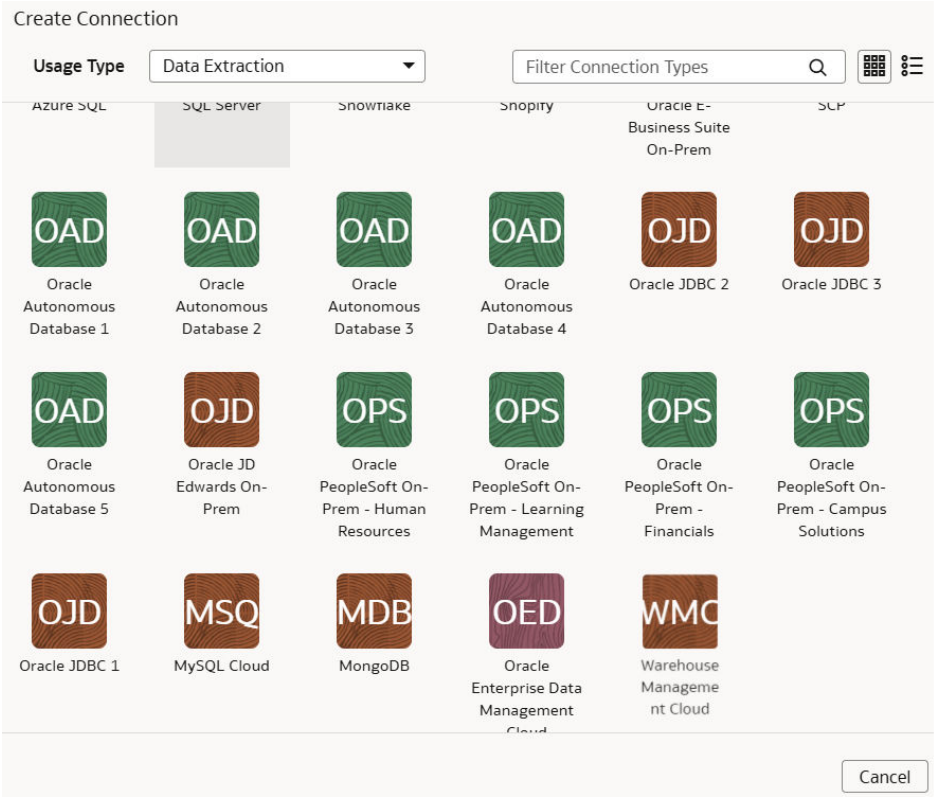
- The Full option has a single date range; StartDate=ExtractDate - NdaysAgo, EndDate=RunDate.
- The Commited option can have multiple date ranges. For each configured GA_LIST_OF_N_DAYS_AGO, StartDate=ExtractDate - NdaysAgo, EndDate=RunDate - NdaysAgo.

Load Data from Mongo Database into NetSuite Analytics Warehouse (Preview)

As a service administrator, you can use the NetSuite Analytics Warehouse extract service to acquire data from the Mongo database and use it to create data augmentations.

Ensure that **MongoDB** is enabled on the Enable Features page prior to creating this connection. See Make Preview Features Available.

1. Check the port number for your Mongo database and create a service request with server host and port details to enable network connectivity to the Mongo database server.
2. In NetSuite Analytics Warehouse, create the Mongo database connection using these instructions:
   a. In NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
   b. On the Data Configuration page, click **Manage Connections**.
   c. On the Manage Connections page, click **Create** and then click **Connection**.
   d. In Create Connection, select **Data Extraction** in **Usage Type** and then select **MongoDB** as the connection type.



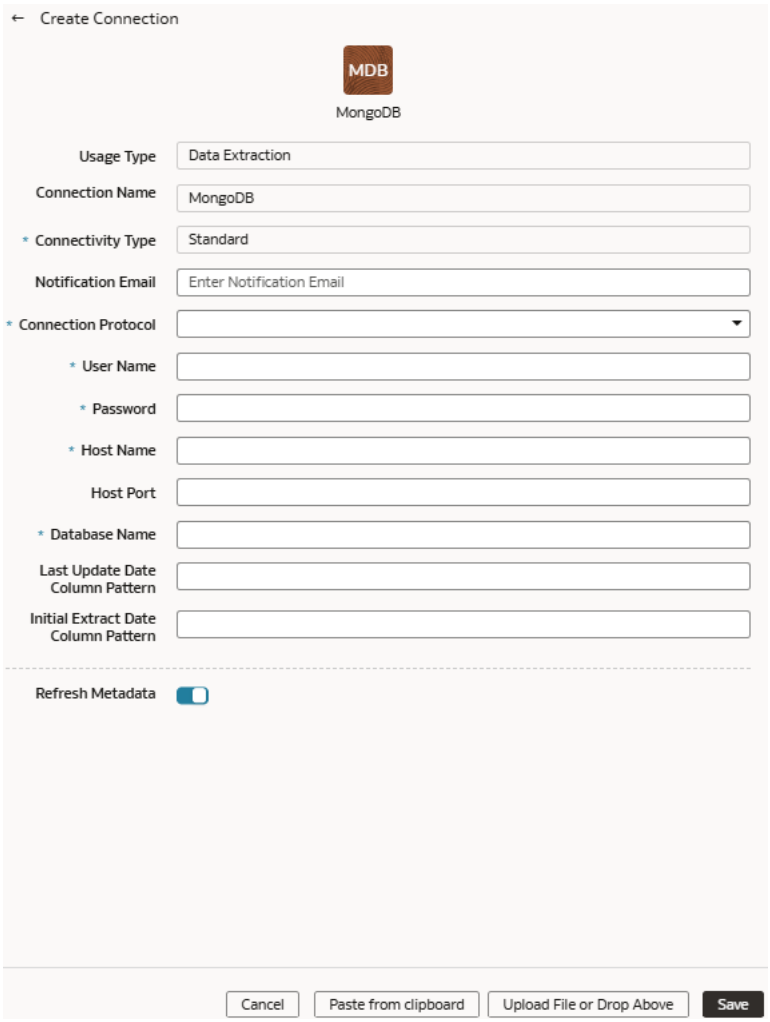
e. In the dialog for the Mongo database connection, select **Standard** in **Connectivity Type**, enter an email address to receive notifications in **Notification Email**, and provide these details:

- **Connection Protocol**: Enter the connection protocol such as `mongodb+srv` or `mongodb`
- Credentials to access the database in **User Name** and **Password**
- **Host Name**: Enter the host name of the Mongo database such as `cluster0.example4.mongodb.net`
- **Host Port**: Enter the port number where Mobgo database is listening such as `27017`
- **Database Name**: Enter a name such as `Analytics`
- **Last Update Date Column Pattern**: Enter pattern such as `"%mo%fie%te%"`

> **Note:**
>
> If column name matches with pattern provided, then NetSuite Analytics Warehouse marks the column name with `isLastUpdateDate = true` and uses it for the incremental extract. For example, if pattern provided is `"%mo%fie%te%"`, then the column name `modifiedDate` is marked as `isLastUpdateDate = true`.

- **Initial Extract Date Column Pattern**: Enter pattern such as `"%cr%ted%te%"`

> **Note:**
>
> If column name matches with pattern provided, then NetSuite Analytics Warehouse marks the column name with `isCreationDate = true` and uses it for the initial extract date extraction. For example, if pattern provided is: `"%cr%ted%te%"`, then the column name `createdDate` is marked as `isCreationDate = true`.



f. Verify that the **Refresh Metadata** toggle is enabled to ensure that the metadata is refreshed when you save the connection.

   You can later refresh the metadata from the Actions menu on the Manage Connections page, if required.

> **Note:**
>
> You can't create augmentations for Mongo database unless you perform a metadata extract.

g. Click **Save**.

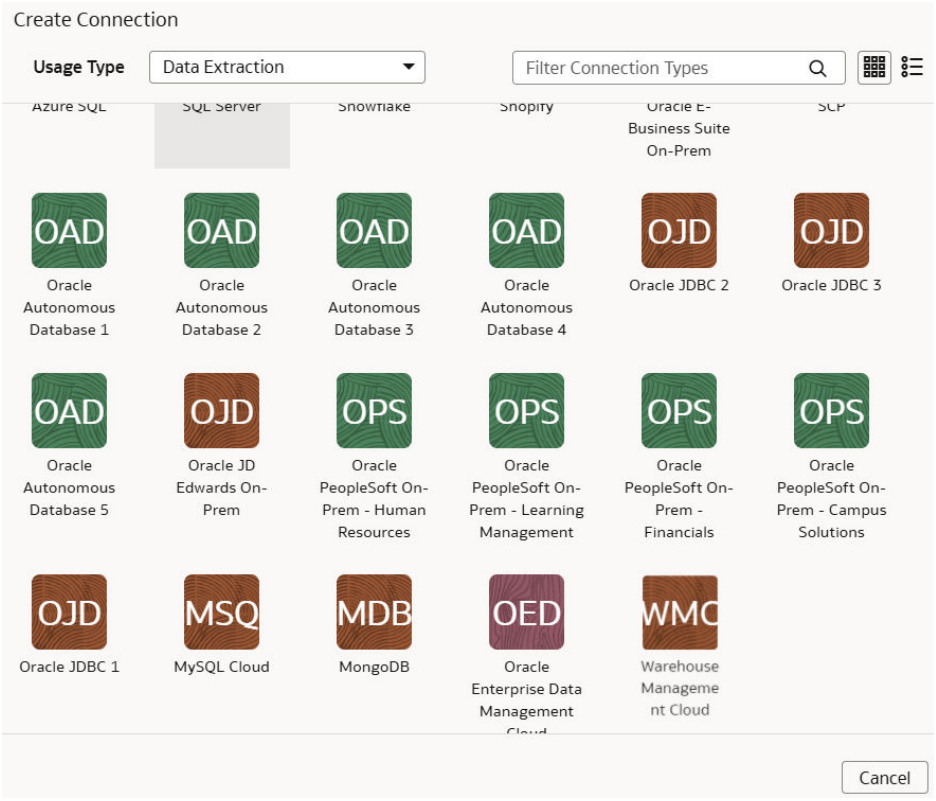
3. On the Manage Connections page, select **Actions** for the Mongo database connection and then select **Test Connection**.
4. After the connections are successfully established, navigate to the Data Configuration page, select the connection in **Data Source**, then click the Data Augmentation tile, and create a data augmentation using the Mongo database data. Select the applicable Mongo database source tables. See Augment Your Data.

Load Data from MySQL Cloud Database into NetSuite Analytics Warehouse (Preview)

As a service administrator, you can use the NetSuite Analytics Warehouse extract service to acquire data from the MySQL Cloud database and use it to create data augmentations.

Ensure that **MySQL Cloud** is enabled on the Enable Features page prior to creating this connection. See Make Preview Features Available.

1. Check the port number for your MySQL Cloud database and create a service request with server host and port details to enable network connectivity to the MySQL server.
2. In NetSuite Analytics Warehouse, create the MySQL Cloud database connection using these instructions:
   a. In NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
   b. On the Data Configuration page, click **Manage Connections**.
   c. On the Manage Connections page, click **Create** and then click **Connection**.
   d. In Create Connection, select **Data Extraction** in **Usage Type** and then select **MySQL Cloud** as the connection type.



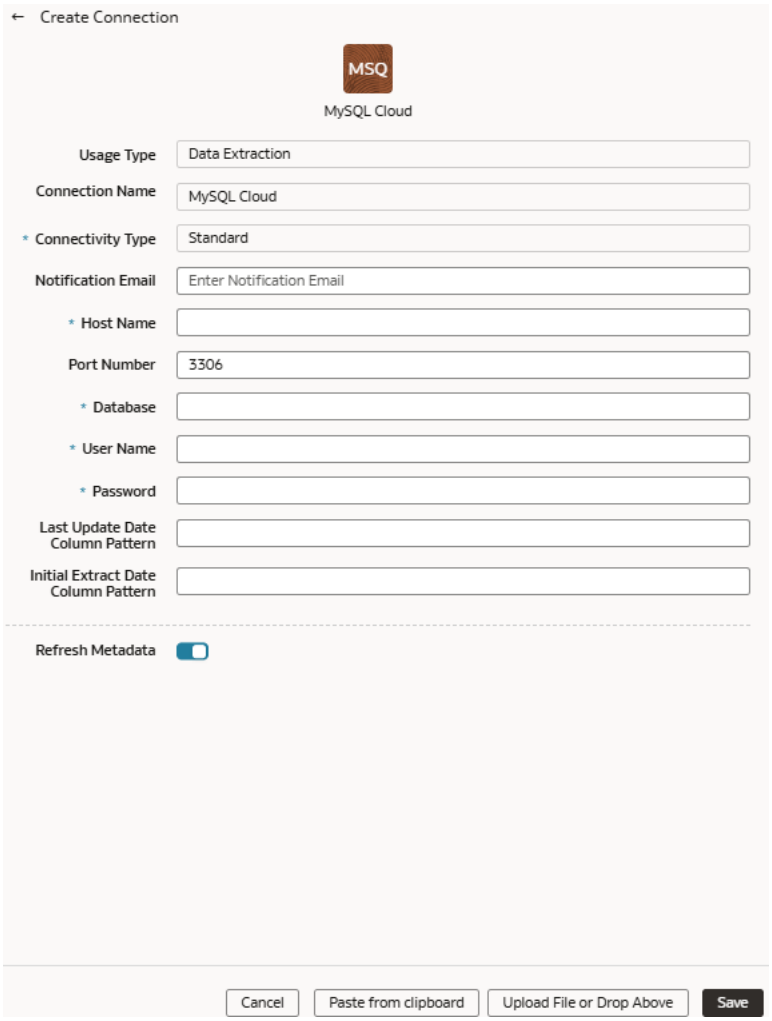
   e. In the dialog for the MySQL Cloud database connection, select **Standard** in **Connectivity Type**, enter an email address to receive notifications in **Notification Email**, and provide these details:
      - **Host Name**: Enter the host name of MySQL server such as `100.111.252.64`

- **Port Number**: Enter the port number where the server is listening such as `3306`
- **Database**: Enter the database name you need to connect to such as `airportdb`
- Credentials to access the database in **User Name** and **Password**
- **Last Update Date Column Pattern**: Enter format such as `"%mo%fie%te%"`

> **Note:**
>
> If column name matches with pattern provided, then NetSuite Analytics Warehouse marks the column name with `isLastUpdateDate = true` and uses it for the incremental extract. For example, if pattern provided is `"%mo%fie%te%"`, then the column name `modifiedDate` is marked as `isLastUpdateDate = true`.

- **Initial Extract Date Column Pattern**: Enter format such as `"%cr%ted%te%"`

> **Note:**
>
> If column name matches with pattern provided, then NetSuite Analytics Warehouse marks the column name with `isCreationDate = true` and uses it for the initial extract date extraction. For example, if pattern provided is: `"%cr%ted%te%"`, then the column name `createdDate` is marked as `isCreationDate = true`.



f. Verify that the **Refresh Metadata** toggle is enabled to ensure that the metadata is refreshed when you save the connection.

   You can later refresh the metadata from the Actions menu on the Manage Connections page, if required.

> **Note:**
>
> You can't create augmentations for MySQL Cloud database unless you perform a metadata extract.

g. Click **Save**.

3. On the Manage Connections page, select **Actions** for the MySQL Cloud database connection and then select **Test Connection**.
4. After the connections are successfully established, navigate to the Data Configuration page, select the connection in **Data Source**, then click the Data Augmentation tile, and create a data augmentation using the MySQL Cloud database data. Select the applicable MySQL Cloud database source tables. See Augment Your Data.

Load Data from Oracle Analytics Publisher into NetSuite Analytics Warehouse (Preview)

As a service administrator, you can use the NetSuite Analytics Warehouse extract service to acquire data from the Analytics Publisher reports and use it to create data augmentations for various use cases.

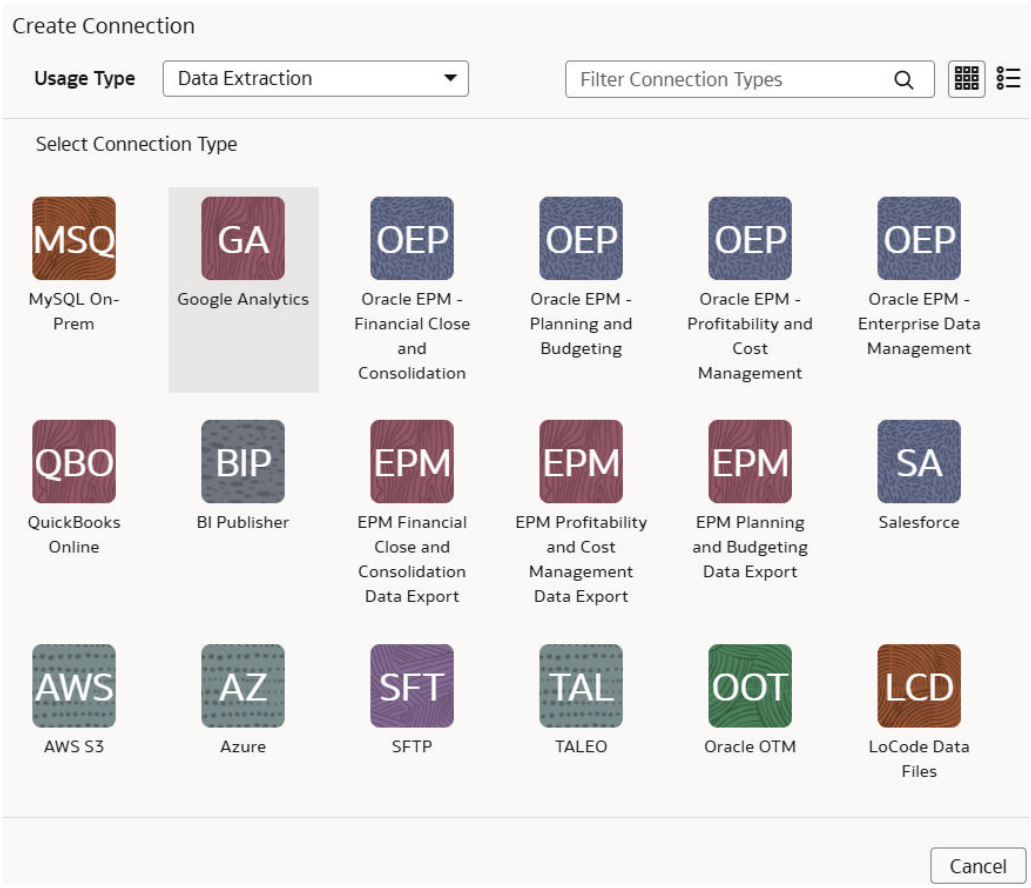
Currently, the BI Publisher Reports connector only supports:

- The Analytics Publisher in Oracle Fusion Cloud Applications for data augmentation.
- Only those reports that complete within the Analytics Publisher report execution timeout limit that's typically 300 seconds.

The BI Publisher Reports connector workflow must observe the security rules of Oracle Fusion Cloud Applications. You must ensure that the password rotation and update are done on time before executing the BI Publisher Reports connector pipeline. Otherwise, those pipeline jobs will hang and eventually those jobs will get deleted, and the data source will be disabled until you update the password and resubmit the job.

Ensure that **BI Publisher Reports** is enabled on the Enable Features page prior to creating this connection. See Make Preview Features Available.

1. In BI Publisher, build a comma separated values (CSV) report for the targeted tables. See Create Reports and Generate CSV Output.
2. In NetSuite Analytics Warehouse, create the data connection to the BI Publisher reports using these instructions:
    a. In NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
    b. On the Data Configuration page, click **Manage Connections**.
    c. On the Manage Connections page, click **Create** and then click **Connection**.
    d. In Create Connection, select **Data Extraction** in **Usage Type** and then select **BI Publisher Reports** as the connection type.



e. In the dialog for the BI Publisher reports connection, provide these details:

- Select **Standard** in **Connectivity Type**.
- Enter an email address to receive notifications in **Notification Email**.
- Host name of the BI Publisher in **Host Name**.
- Credentials to access your BI Publisher in **User Name** and **Password**
- Enter comma separated reports in **List of Reports** in the `<reports directory>/<report names>` format.

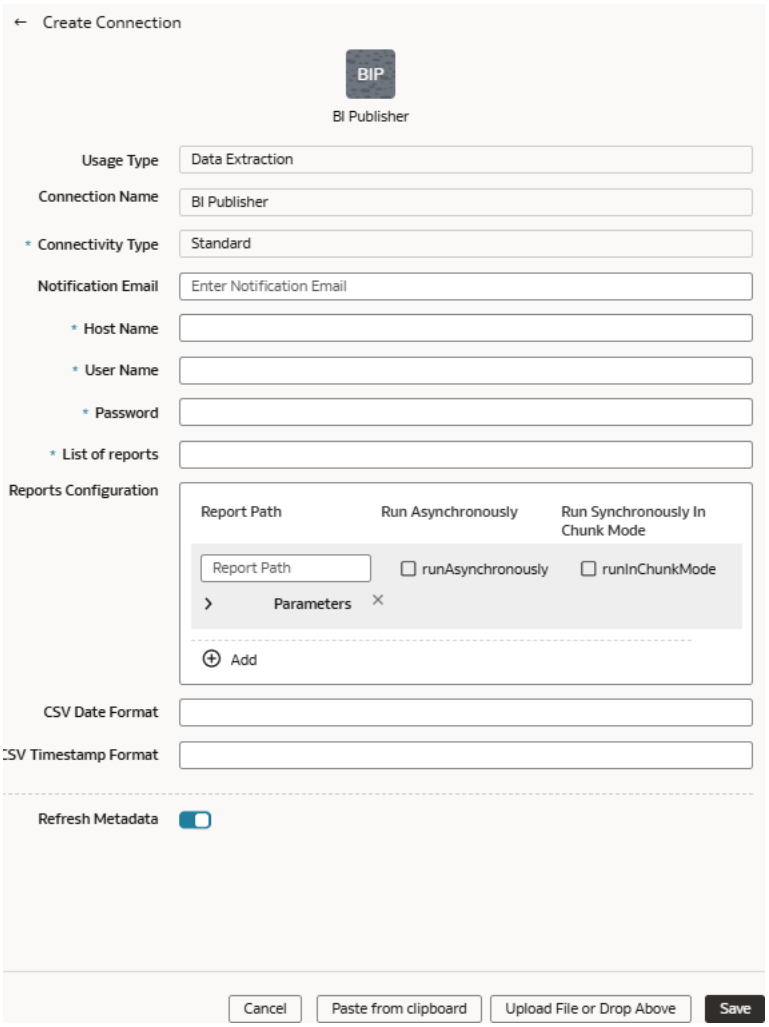
> **Note:**
>
> Ensure that the reports directory and report names don't have spaces.

- In **Reports Configuration**, provide the path of the report in Oracle Analytics Publisher, select **Run Asynchronously** if the report isn't huge, else select **Run Synchronously In Chunk** if the report has a large volume of data (more than 7 MB in size) because this enables you to download the report in chunks of data.

> **Note:**
>
> – If the report execution is completed before the timeout limit and generates large volume of data then set `runInChunkMode: true`. The recommendation is to set `runInChunkMode` to true if the report output file is over 7MB.
>
> – If the report has parameters defined, then provide the values in the "params" section of the report configuration array. If the value in the params array is set with placeholder `__lastUpdateDate__`, then the connector applies the initial extract date or last successful extract date to the param.

- For **CSV Date Format** and **CSV Timestamp Format**, see About Date and Timestamp Formatting for CSV File-based Extractors.

> **Note:**
>
> Ensure that the date format used in Oracle Analytics Publisher and NetSuite Analytics Warehouse match.



f. Verify that the **Refresh Metadata** toggle is enabled to ensure that the metadata is refreshed when you save the connection.

   You can later refresh the metadata from the Actions menu on the Manage Connections page, if required.

> **Note:**
>
> You can't create augmentations for the BI Publisher reports unless you perform a metadata extract.

g. Click **Save**.

3. On the Manage Connections page, select **Actions** for the BI Publisher reports connection and then select **Test Connection**.
4. After the connections are successfully established, navigate to the Data Configuration page, select the connection in **Data Source**, then click the Data Augmentation tile, and create a data augmentation using the BI Publisher reports data. Select the applicable BI Publisher source tables. See Augment Your Data.

Load Data from Oracle Database Using JDBC into NetSuite Analytics Warehouse (Preview)

As a service administrator, you can use an extract service remote agent to connect to an Oracle database using JDBC and use the data to create data augmentations.

After connecting to an Oracle database using JDBC, the remote agent extracts the data and loads it into the autonomous data warehouse associated with your Oracle NetSuite Analytics Warehouse instance. The remote agent pulls the metadata through the public extract service REST API and pushes data into object storage using the object storage REST API. You can extract and load the data from an Oracle database into Oracle NetSuite Analytics Warehouse only once every 24 hours.
Ensure that **Remote Agent** and **Oracle JDBC** are enabled on the Enable Features page prior to creating this connection. See Make Preview Features Available.

1. Set up the remote agent to load data from your SQL Server into Oracle NetSuite Analytics Warehouse.

   See Set up the Remote Agent to Load Data into NetSuite Analytics Warehouse (Preview).

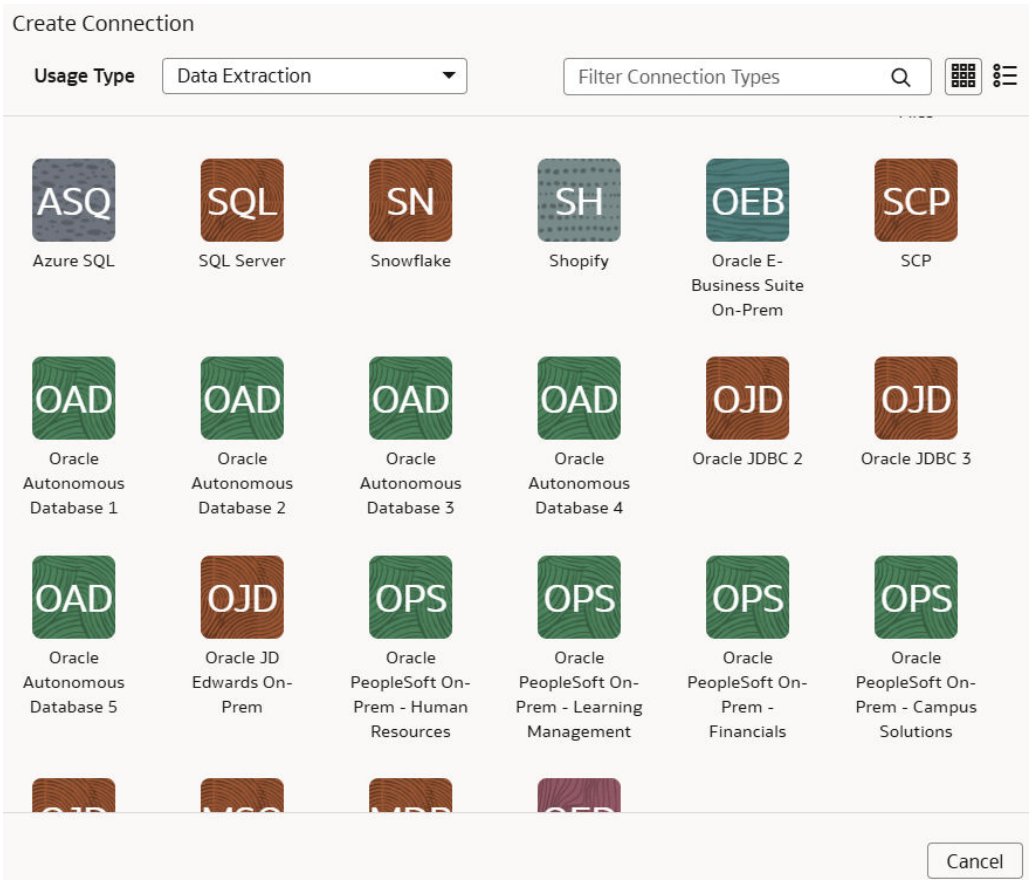
2. Configure the remote agent and Oracle database data source on the Data Configuration page in Oracle NetSuite Analytics Warehouse using these instructions:

   a. On the **Console**, click **Data Configuration** under **Application Administration**.

   b. On the Data Configuration page, click **Manage Connections**.

   c. On the Manage Connections page, click **Create** and then click **Connection**.

   d. In Create Connection, select **Data Extraction** in **Usage Type**, and then select **Oracle JDBC** as the connection type.



e. In Create Connection for Oracle JDBC, in **Connectivity Type**, verify that **Remote** is selected automatically.

f. In **Remote Agent**, select the remote agent connection that you created earlier, for example, **Remote Agent**.

g. Enter an email address to receive notifications in **Notification Email**, provide credentials for the Oracle database source in **User Name** and **Password**, and the URL of the Oracle database source in **URL**.

h. In **Initial Extract Date Column Pattern**, provide the initial extract date pattern that matches the pattern in your source.

i. In **Last Update Date Column Pattern**, provide the last update date pattern that matches the pattern in your source.

j. If your source has flashback support, then select **Yes** in **Enable flashback incremental**.

k. In **List of Flashback Incremental Exempt datastores**, provide a comma separated list of datastores that you want to exempt from the flashback incremental queries.

l. If your source has ROWSCN support, then select **Yes** in **Enable ROWSCN incremental**.

m. In **List of ROWSCN Incremental Exempt datastores**, specify a comma-separated list of datastores that you want to exclude from the automatic tracking of row changes based on system change numbers.

n. In **Case Sensitive Data Stores**, select Yes or No to specify whether the datastores have case sensitive data.

o. In **Schema Name**, enter the schema name to extract data from.

p. In **Data Store Name Pattern**, specify the name pattern of the datastores that you want extracted. If you provide this value, then the connector extracts only datastores matching the pattern.

q. Verify that the **Refresh Metadata** toggle is enabled to ensure that the metadata is refreshed when you save the connection. You can later refresh the metadata from the Actions menu on the Manage Connections page, if required.

> **Note:**
>
> You can't create augmentations for Oracle database unless you perform a metadata extract.

r. Click **Save**.

3. After the connections are successfully established, navigate to the Data Configuration page, select the connection in **Data Source**, then click the Data Augmentation tile, and create a data augmentation using the Oracle database data. Select the applicable Oracle database source tables. See Augment Your Data.

Load Data from Salesforce into NetSuite Analytics Warehouse

As a service administrator, you can use the NetSuite Analytics Warehouse extract service to acquire data from the Salesforce SaaS instance and use it to create data augmentations.

1. Run the following command in your terminal to generate a keystore file:

```
openssl pkcs12 -export -inkey server.key -in server.crt -name
YOUR_KEY_NAME -passout pass:YOUR_PASSWORD -out keystorefile.p12
```

Replace YOUR_KEY_NAME and YOUR_PASSWORD with your desired values, and save the generated `keystorefile.p12` file to later upload it while creating the connection to your Salesforce source.

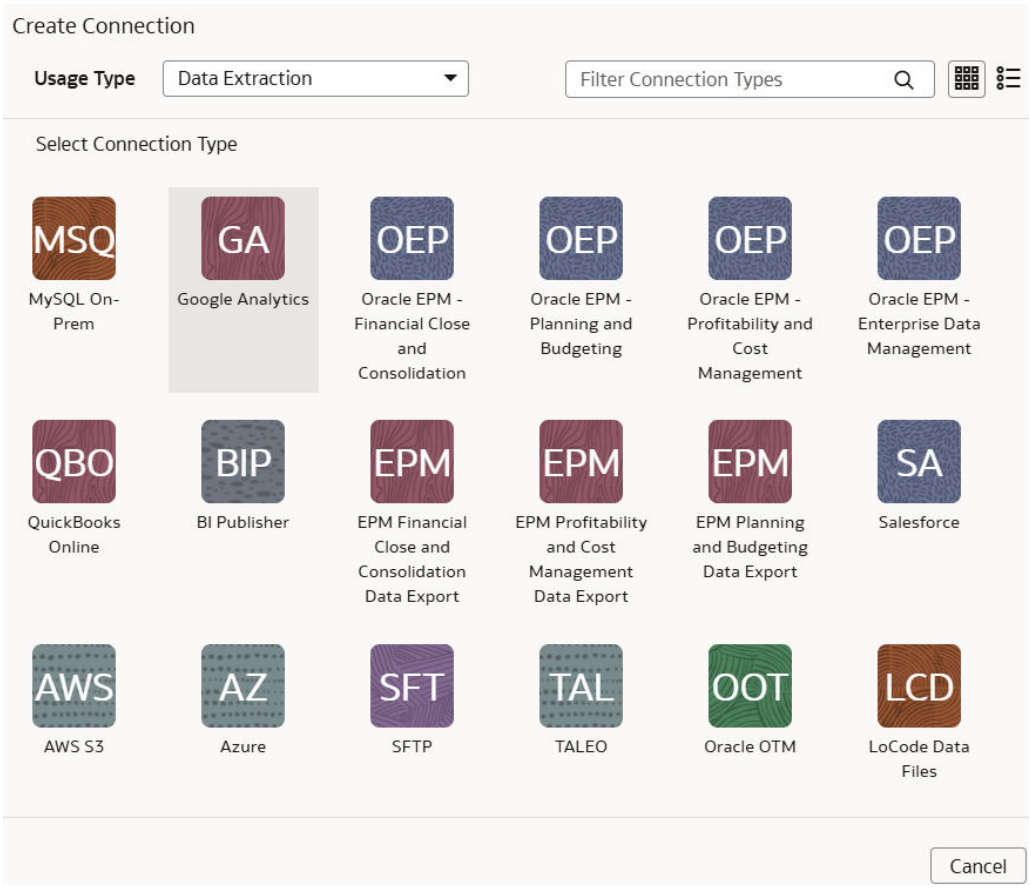
2. In NetSuite Analytics Warehouse, create the Salesforce data connection using these instructions:

a. In NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.

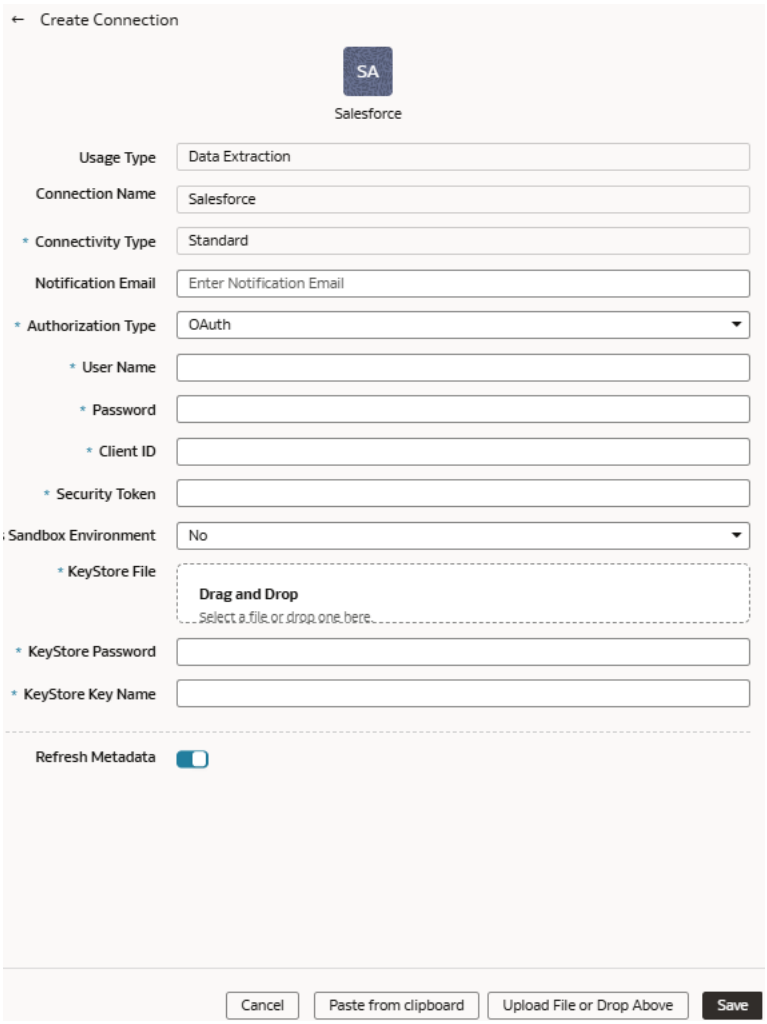
b. On the Data Configuration page, click **Manage Connections**.

c. On the Manage Connections page, click **Create** and then click **Connection**.

d. In Create Connection, select **Data Extraction** in **Usage Type** and then select **Salesforce** as the connection type.



e. In the dialog box for the Salesforce connection, select **Standard** in **Connectivity Type** and enter an email address to receive notifications in **Notification Email**.



f. In **Authorization Type**, you can select **Basic Authentication** or **OAuth** as the type of authorization.

   If you select Basic Authentication, then:

   - Enter the credentials for your Salesforce source in **User Name** and **Password**. The user stated here must have access to all the data in the Salesforce system to extract it to the warehouse.
   - Copy and paste the security token from your Salesforce account in **Security Token**. This is an alpha-numeric code and may contain special characters, however, it isn't visible. It's encrypted and shown as ….
   - In **Is Sandbox Environment**, select **Yes** if your Salesforce source is a test or sandbox environment; else select **No**.

   If you select **OAuth**, then enter these additional values:

   - In **Client ID**, enter the unique identifier that represents the Salesforce application within the Salesforce instance and is visible when you sign into your Salesforce account.
   - In **KeyStore File**, provide the file that is in a PKCS#12 file format (denoted by .p12 extension), which can contain both private keys and certificates (such as the public key of the server or the signing certificate).

- In **KeyStore Password**, enter the password for accessing the specified keystore file.
- In **Keystore Key name**, provide the unique name (identifier or alias) of the specified keystore file.

g. Verify that the **Refresh Metadata** toggle is enabled to ensure that the metadata is refreshed when you save the connection.

You can later refresh the metadata from the Actions menu on the Manage Connections page, if required.

> **Note:**
>
> You can't create augmentations for Salesforce unless you perform a metadata extract.

h. Click **Save**.

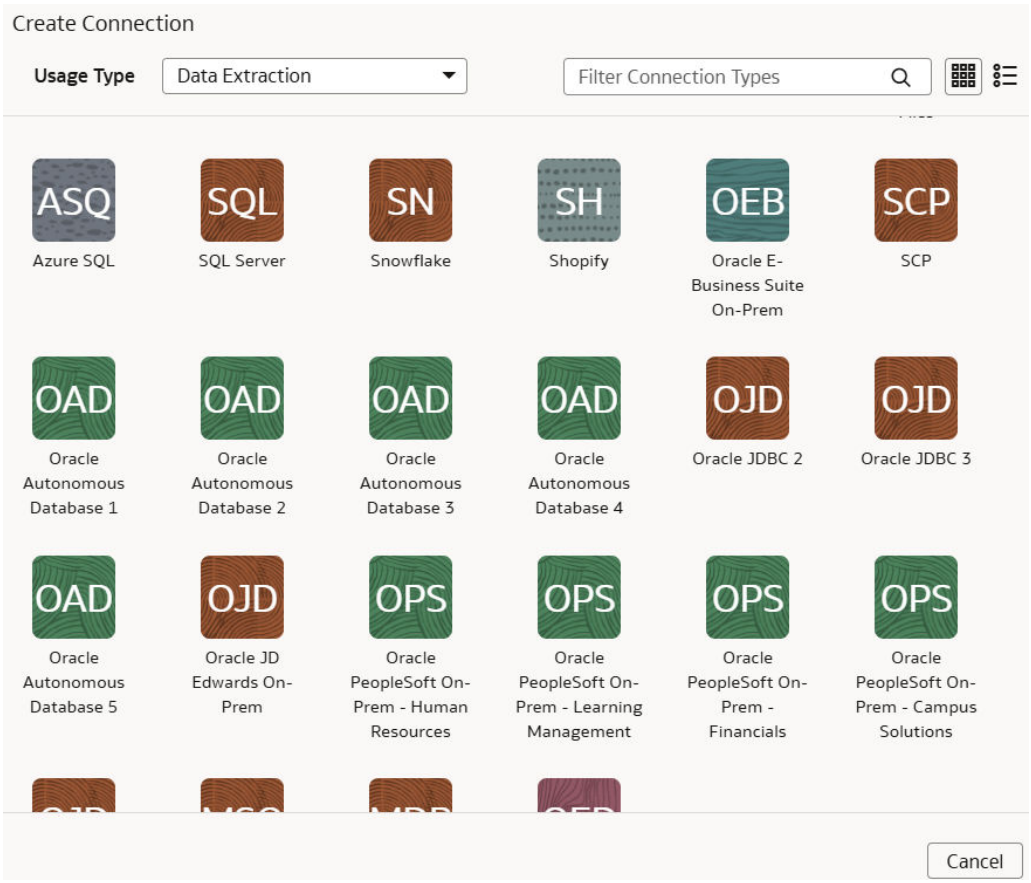
3. On the Manage Connections page, select **Actions** for the Salesforce connection and then select **Test Connection**.
4. After the connections are successfully established, navigate to the Data Configuration page, select the connection in **Data Source**, then click the Data Augmentation tile, and create a data augmentation using the Salesforce data. Select the applicable Salesforce source tables. See Augment Your Data.

Load Data from Shopify into NetSuite Analytics Warehouse (Preview)

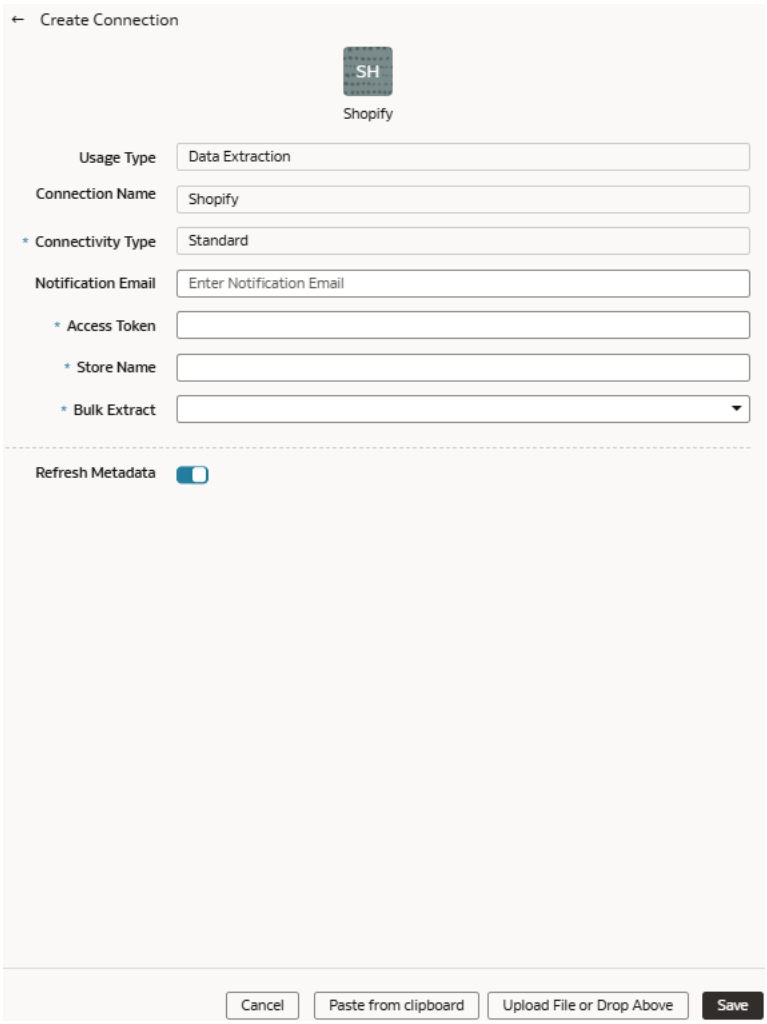
As a service administrator, you can use the NetSuite Analytics Warehouse extract service to acquire data from the Shopify SaaS instance and use it to create data augmentations for various Enterprise Resource Planning and Supply Chain Management use cases.

Ensure that **Shopify** is enabled on the Enable Features page prior to creating this connection. See Make Preview Features Available.

1. In NetSuite Analytics Warehouse, create the Shopify data connection using these instructions:
   a. In NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
   b. On the Data Configuration page, click **Manage Connections**.
   c. On the Manage Connections page, click **Create** and then click **Connection**.
   d. In Create Connection, select **Data Extraction** in **Usage Type** and then select **Shopify** as the connection type.



e. In the dialog for the Shopify connection, select **Standard** in **Connectivity Type**, enter an email address to receive notifications in **Notification Email**, applicable token value in **Access Token**, **Store Name** such as myfawteststore.myshopify.com, and `True` in **Bulk Extract**.



f. Verify that the **Refresh Metadata** toggle is enabled to ensure that the metadata is refreshed when you save the connection.

   You can later refresh the metadata from the Actions menu on the Manage Connections page, if required.

> **Note:**
>
> You can't create augmentations for Shopify unless you perform a metadata extract.

g. Click **Save**.

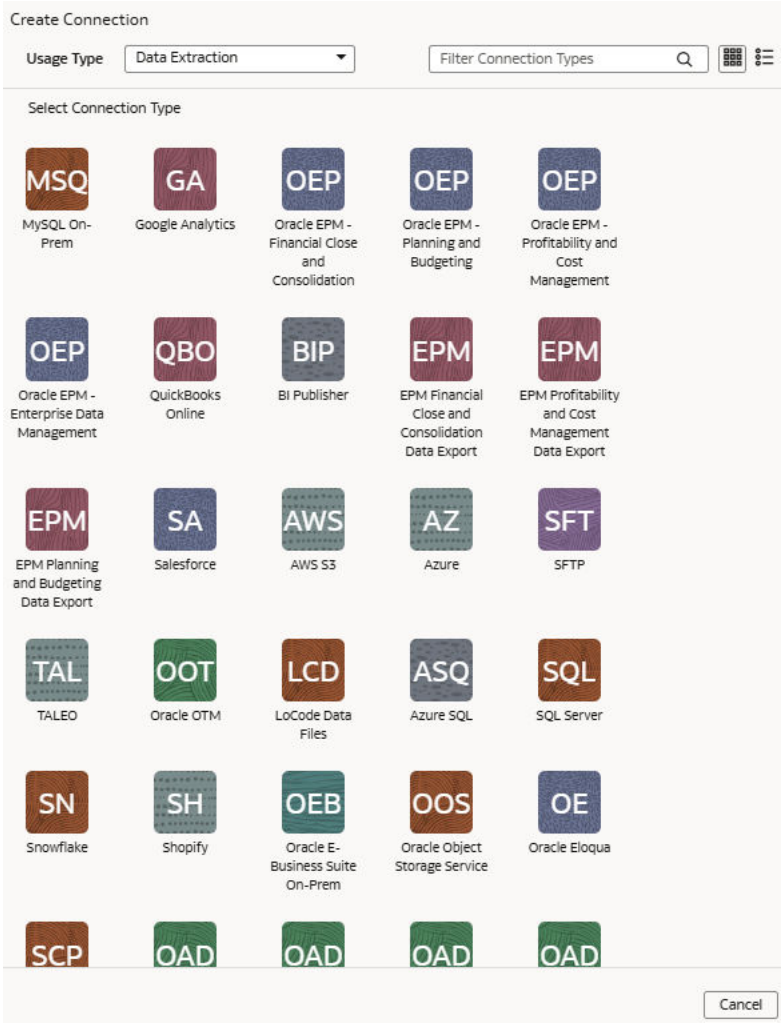
2. On the Manage Connections page, select **Actions** for the Shopify connection and then select **Test Connection**.
3. After the connections are successfully established, navigate to the Data Configuration page, select the connection in **Data Source**, then click the Data Augmentation tile, and create a data augmentation using the Shopify data. Select the applicable Shopify source tables. See Augment Your Data.

Connect with Your Oracle Eloqua Data Source (Preview)

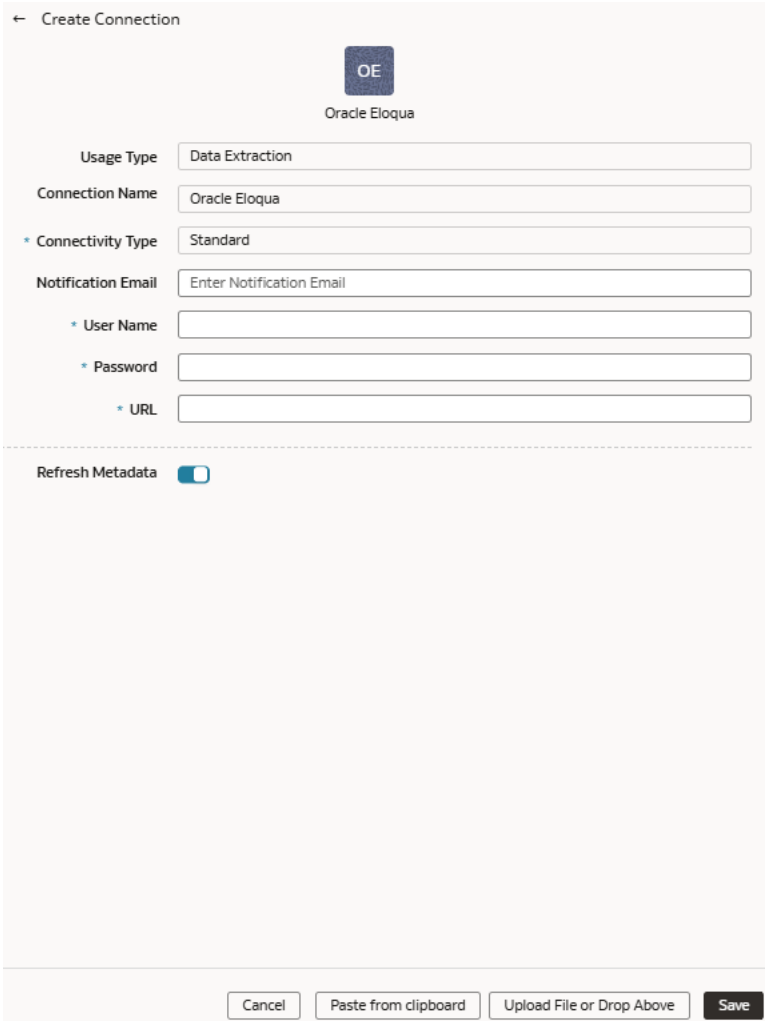
If you've subscribed for Oracle Fusion CX Analytics and want to load data from your Oracle Eloqua source into NetSuite Analytics Warehouse, then create a connection using the *Eloqua* connection type.

The Oracle Eloqua data that you load into NetSuite Analytics Warehouse enables you to augment the data in your warehouse and create varied customer experience-related analytics. Ensure that **Oracle Eloqua** is enabled on the Enable Features page prior to creating this connection. See Make Preview Features Available.

1. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
2. On the Data Configuration page, click **Manage Connections** under Global Configurations.
3. On the Manage Connections page, click **Create** and then click **Connection**.
4. In Create Connection, , select **Data Extraction** in **Usage Type**, and then select **Oracle Eloqua** as the connection type.



5. In the dialog for the Eloqua connection, select **Standard** in **Connectivity Type**, enter an email address to receive notifications in **Notification Email**, and the credentials to connect with the Eloqua source in **User Name** and **Password**.
6. In **URL**, enter the URL for your Eloqua server in this sample format: `https://<your eloqua server>/api/odata`.



7. Verify that the **Refresh Metadata** toggle is enabled to ensure that the metadata is refreshed when you save the connection.

   You can later refresh the metadata from the Actions menu on the Manage Connections page, if required.

> **Note:**
>
> You can't create augmentations for Eloqua unless you perform a metadata extract.

8. Click **Save**.
9. On the Manage Connections page, select **Actions** for the Eloqua connection and then select **Test Connection**.
10. After the connections are successfully established, navigate to the Data Configuration page, select the connection in **Data Source**, then click the Data Augmentation tile, and

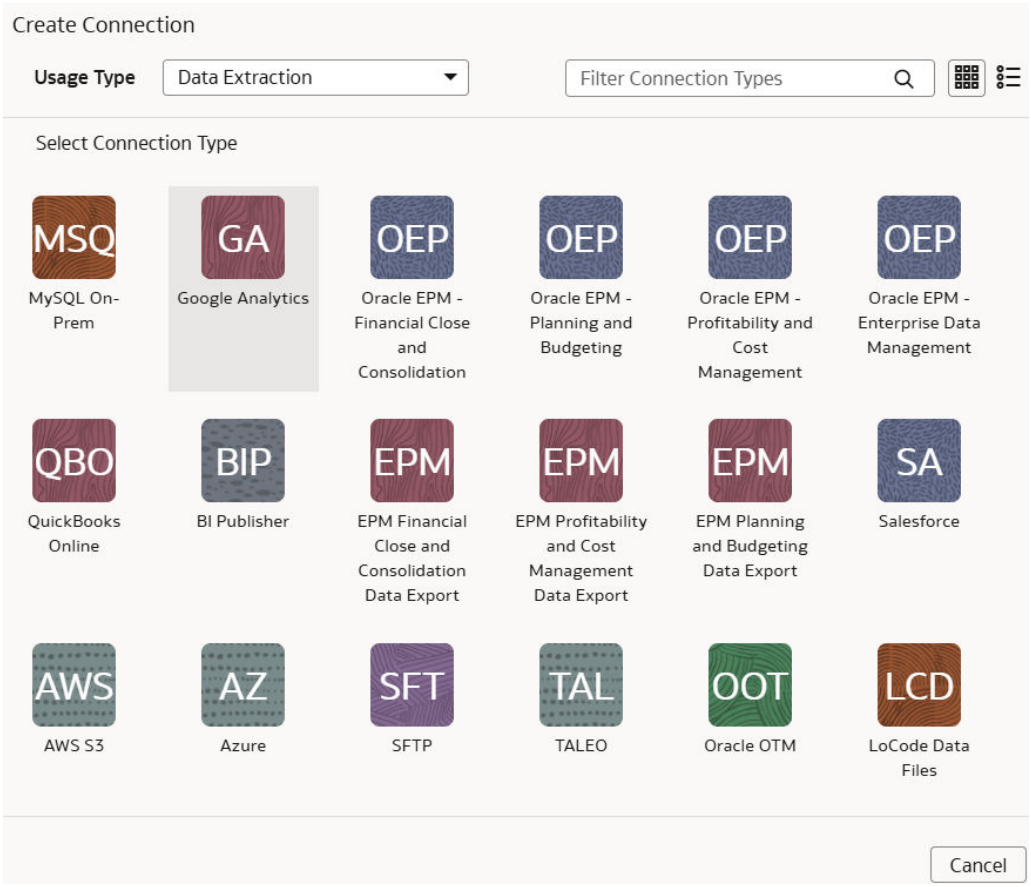
create a data augmentation using the Eloqua data. Select the applicable Eloqua source tables. See Augment Your Data.

Load Data from QuickBooks Online into Oracle NetSuite Analytics Warehouse (Preview)

As a service administrator, you can use the Oracle NetSuite Analytics Warehouse extract service to acquire data from QuickBooks Online and use it to create data augmentations.

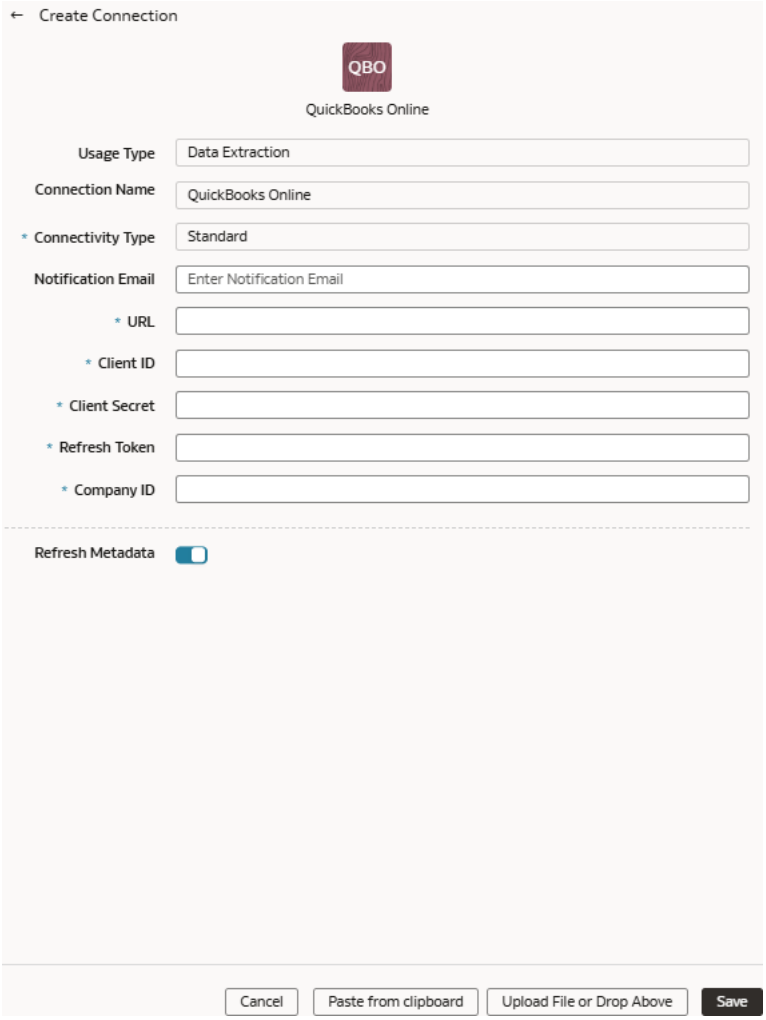
Ensure that **QuickBooks Online** is enabled on the Enable Features page prior to creating this connection. See Make Preview Features Available.

1. In Oracle NetSuite Analytics Warehouse, create the QuickBooks Online data connection using these instructions:
   a. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
   b. On the Data Configuration page, click **Manage Connections**.
   c. On the Manage Connections page, click **Create** and then click **Connection**.
   d. In Create Connection, select **Data Extraction** in **Usage Type** and then select **QuickBooks Online** as the connection type.



   e. In the dialog for the QuickBooks Online connection, enter these details and click **Save**:
      - **Connectivity Type**: Select **Standard**.

- **Notification Email**: Enter an email address to receive notifications.
- **URL**: Enter the complete URL of your Quickbooks Online instance.
- **Client ID**: This is the unique ID on your Quickbooks Online instance.
- **Client Secret**: Copy and paste the entire "client secret" from your authentication.
- **Refresh Token**: This token is changed by Quickbooks everyday; enter your refresh token for the day you want to execute this pipeline.
- **Company ID**: This is your company ID for the Quickbooks Online instance.



f. Verify that the **Refresh Metadata** toggle is enabled to ensure that the metadata is refreshed when you save the connection.

You can later refresh the metadata from the Actions menu on the Manage Connections page, if required.

> **Note:**
>
> You can’t create augmentations for QuickBooks Online unless you perform a metadata extract.

g. Click **Save**.

2. On the Manage Connections page, select **Actions** for the QuickBooks Online connection and then select **Test Connection**.
3. After the connections are successfully established, navigate to the Data Configuration page, select the connection in **Data Source**, then click the Data Augmentation tile, and create a data augmentation using the QuickBooks Online data. Select the applicable QuickBooks Online source tables. See Augment Your Data.

Load Data from Snowflake into Oracle NetSuite Analytics Warehouse (Preview)

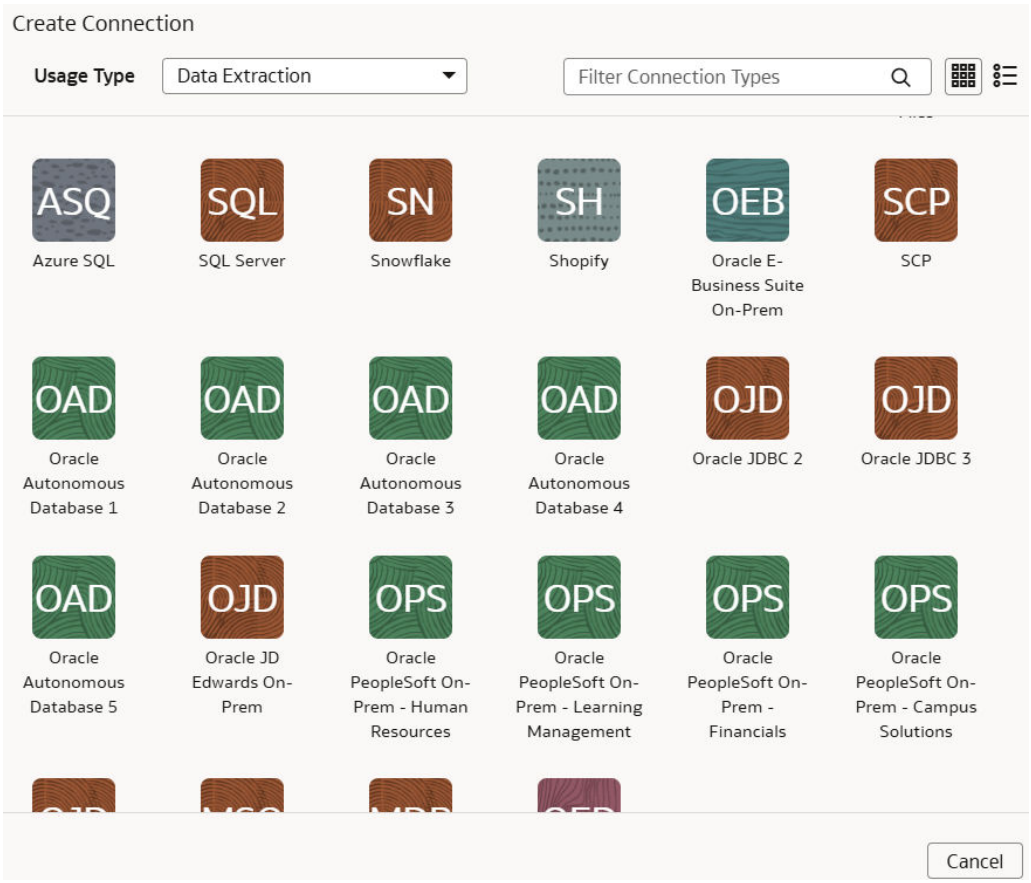
As a service administrator, you can use the Oracle NetSuite Analytics Warehouse extract service to acquire data from a Snowflake instance.

You can later use this data to create data augmentations for various Enterprise Resource Planning and Supply Chain Management use cases. Establish the connection from NetSuite Analytics Warehouse to your Snowflake instance to start data acquisition followed by augmentation.

> **Note:**
>
> Snowflake some times requires API calls to originate from a known IP address. If you're experiencing connection issues due to an unauthorized IP, then submit an Oracle Support ticket to obtain the necessary Oracle IP address for your Snowflake allowlist.

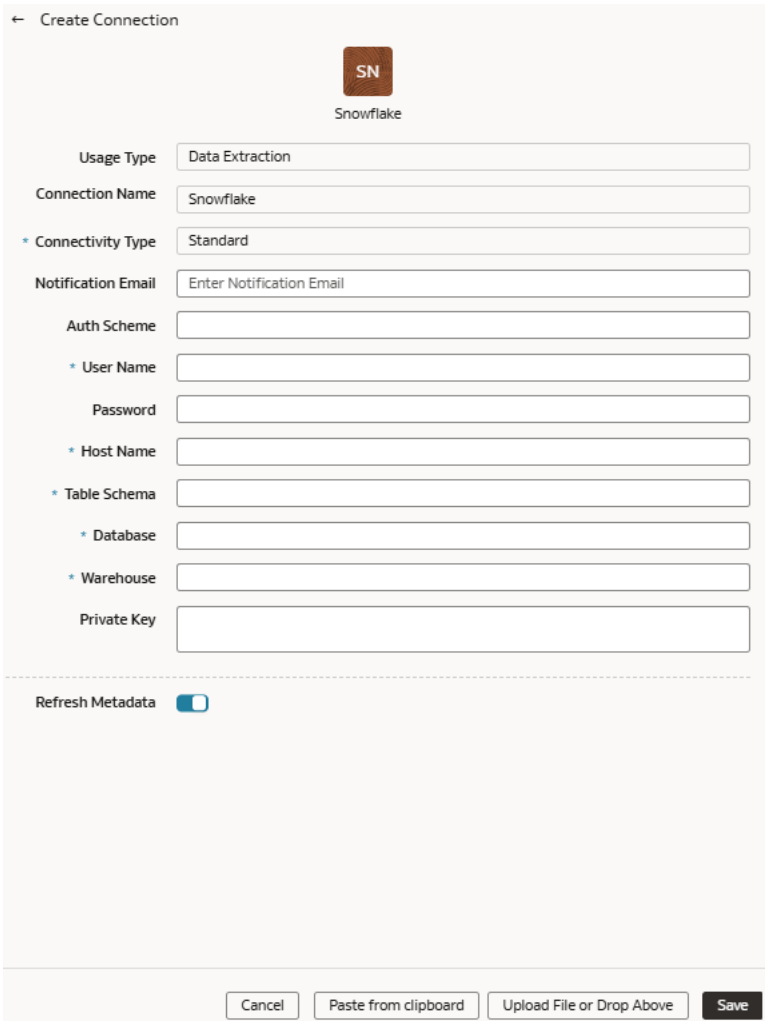
Ensure that **Snowflake** is enabled on the Enable Features page prior to creating this connection. See Make Preview Features Available.

1. In NetSuite Analytics Warehouse, create the Snowflake data connection:
   a. In NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
   b. On the Data Configuration page, click **Manage Connections**.
   c. On the Manage Connections page, click **Create** and then click **Connection**.
   d. In Create Connection, select **Data Extraction** in **Usage Type** and then select **Snowflake** as the connection type.



e. In **Create Connection**, enter these details and then click **Save**:

- **Connectivity Type**: Standard.
- **Notification Email**: An email address to receive notifications.
- **Auth Schema**: Enter “BASICAUTH” if you’re using username and password to establish the connection. Enter “PRIVATE_KEY” if you’re using token-based authentication.
- **User Name**: Enter username only if you're using the basic authentication.
- **Password**: Enter password for the username only if you're using the basic authentication.
- **Host Name**: Complete host name of your Snowflake instance.
- **Table Schema**: Your Snowflake table schema such as TPCH_SF1.
- **Database**: Mentioned in your Snowflake account under **Data**.
- **Warehouse**: The compute resources in your Snowflake instance that you can find by running SHOW WAREHOUSES [LIKE '<*pattern*>']. See SHOW WAREHOUSES.
- **Private Key**: Generate the Private Key in Snowflake, if you don’t have one already, and paste it here. See Generate the Private Key.



f. Verify that the **Refresh Metadata** toggle is enabled to ensure that the metadata is refreshed when you save the connection.

   You can later refresh the metadata from the Actions menu on the Manage Connections page, if required.

> **Note:**
>
> You can't create augmentations for Snowflake unless you perform a metadata extract.

g. Click **Save**.

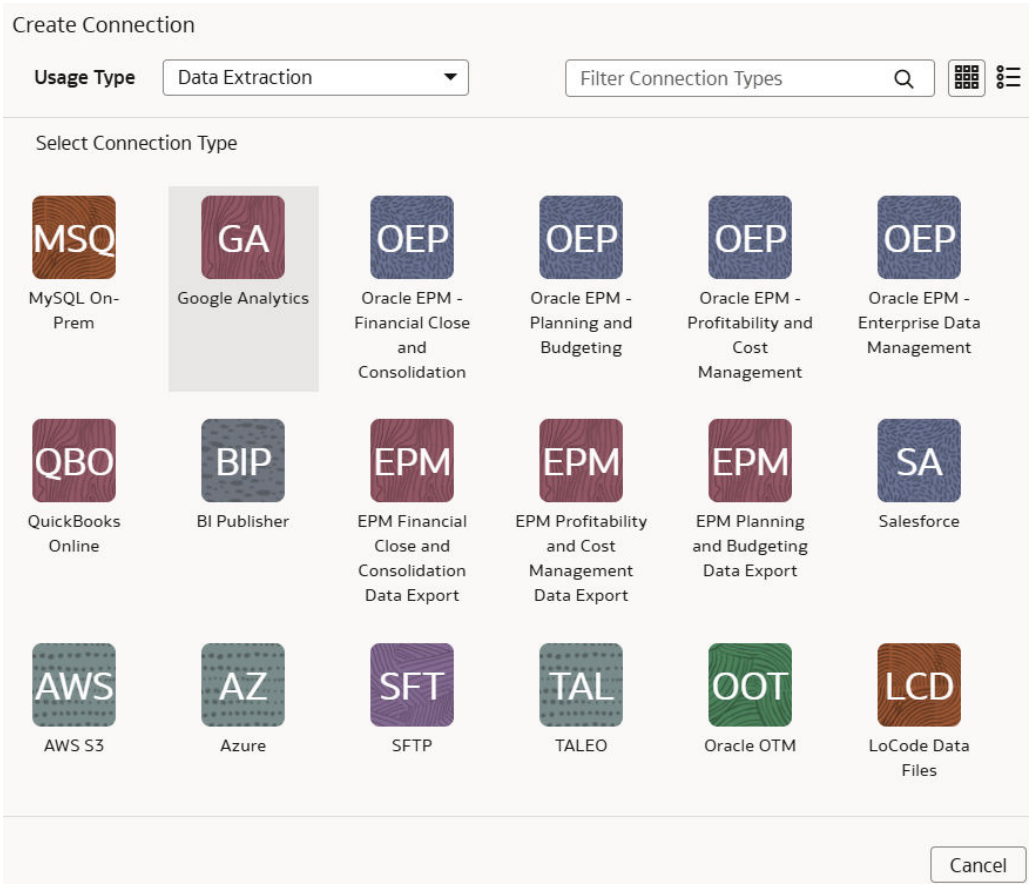
2. On the Manage Connections page, select **Actions** for the Snowflake connection and then select **Test Connection**.
3. After the connections are successfully established, navigate to the Data Configuration page, select the connection in **Data Source**, then click the Data Augmentation tile, and create a data augmentation using the Snowflake data. Select the applicable Snowflake source tables. See Augment Your Data.

Load Data from Taleo into NetSuite Analytics Warehouse (Preview)

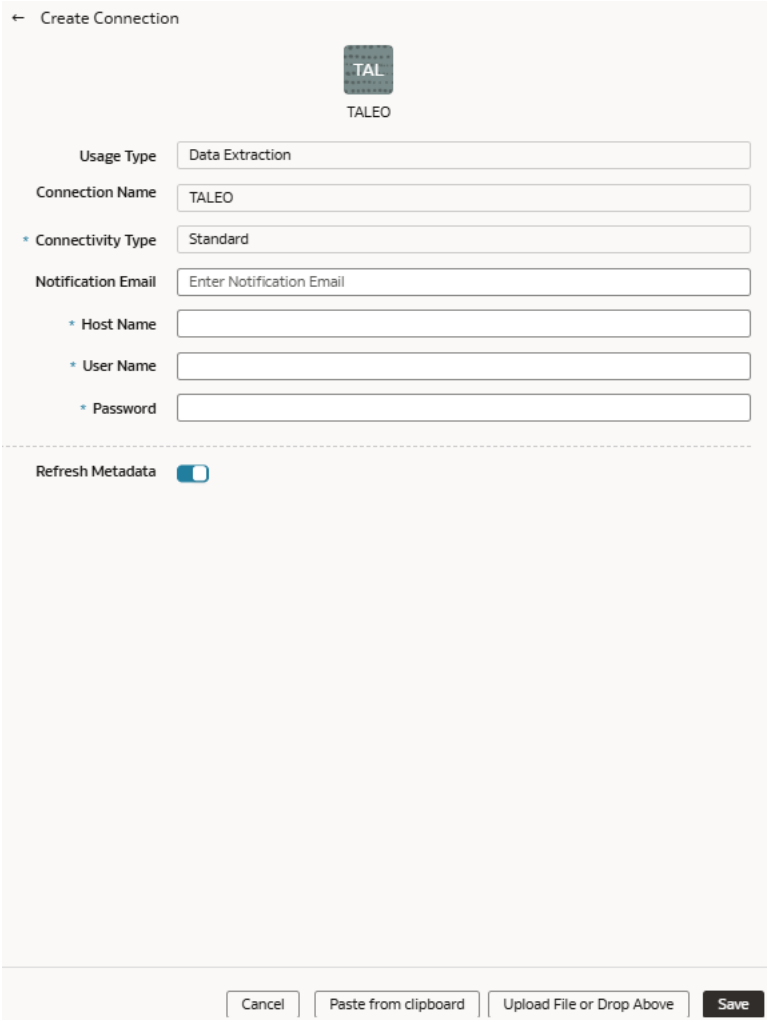
As a service administrator, you can use the NetSuite Analytics Warehouse extract service to acquire data from the Taleo instance and use it to create data augmentations for various Enterprise Resource Planning and Supply Chain Management use cases.

Ensure that **Taleo** is enabled on the Enable Features page prior to creating this connection. See Make Preview Features Available.

1. In NetSuite Analytics Warehouse, create the Taleo data connection using these instructions:
   a. In NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
   b. On the Data Configuration page, click **Manage Connections**.
   c. On the Manage Connections page, click **Create** and then click **Connection**.
   d. In Create Connection, select **Data Extraction** in **Usage Type** and then select **Taleo** as the connection type.



   e. In **Connectivity Type**, select **Standard**, enter an email address to receive notifications in **Notification Email**, host name of your Taleo instance in **Host Name**, and credentials for your Taleo source in **User Name** and **Password**.



f. Verify that the **Refresh Metadata** toggle is enabled to ensure that the metadata is refreshed when you save the connection.

   You can later refresh the metadata from the Actions menu on the Manage Connections page, if required.

> **Note:**
>
> You can't create augmentations for Taleo unless you perform a metadata extract.

g. Click **Save**.

2. On the Manage Connections page, select **Actions** for the Taleo connection and then select **Test Connection**.
3. After the connections are successfully established, navigate to the Data Configuration page, select the connection in **Data Source**, then click the Data Augmentation tile, and create a data augmentation using the Taleo data. Select the applicable Taleo source tables. See Augment Your Data. See Augment Your Data.

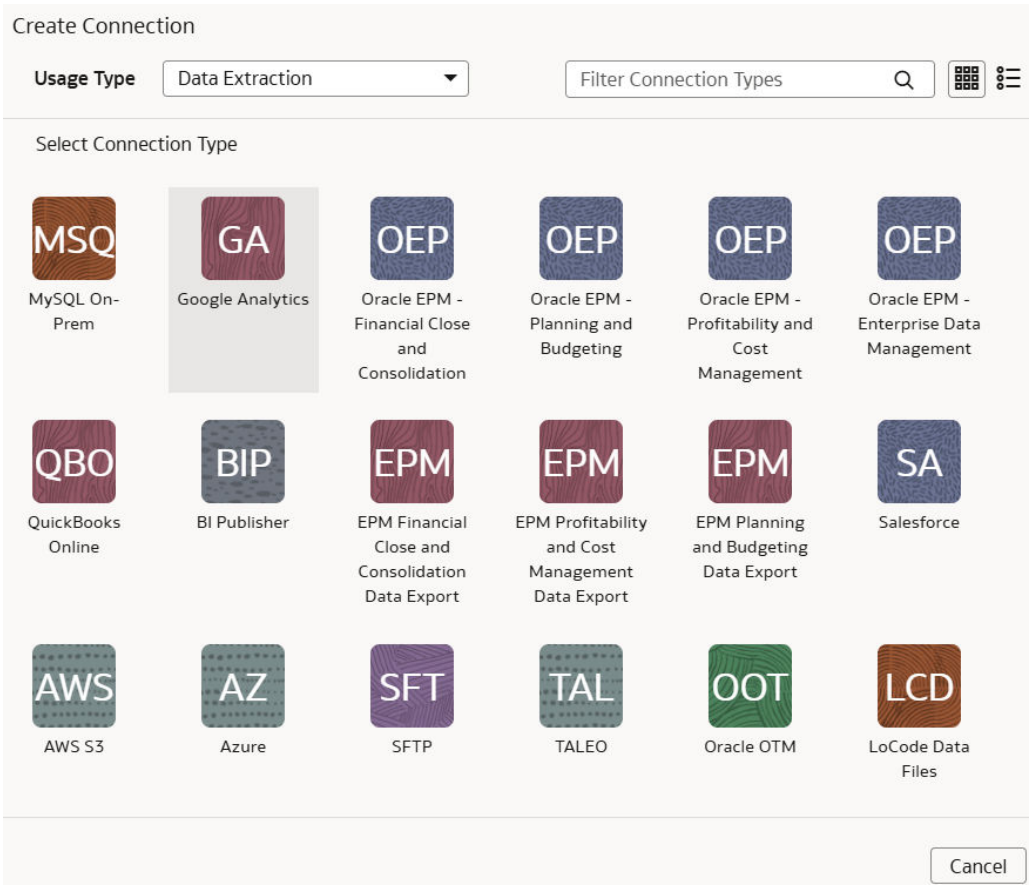
Load Data from Oracle Transportation Management Cloud Service into Oracle NetSuite Analytics Warehouse (Preview)

As a service administrator, you can use the Oracle NetSuite Analytics Warehouse extract service to acquire data from an Oracle Transportation Management Cloud Service SaaS instance.

You can later use this data to create data augmentations for various Enterprise Resource Planning and Supply Chain Management use cases. Establish the connection from NetSuite Analytics Warehouse to your Oracle Transportation Management Cloud Service instance to start data acquisition followed by augmentation.

> **Note:**
>
> Oracle Fusion SCM Analytics is a prerequisite to use the "Oracle Transportation Management" connector.

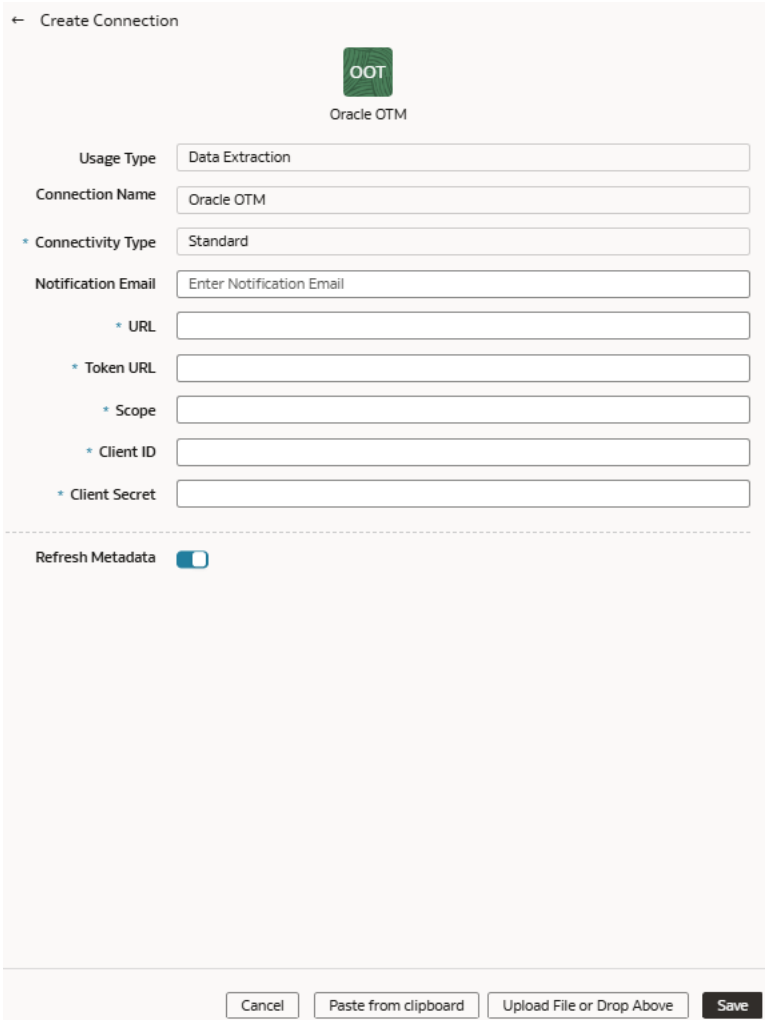
1. Create a confidential application in Oracle Identity Cloud Service. See IDCS Configuration.
2. In Oracle Transportation Management Cloud Service, create a user and assign the "Integration" role using these instructions:
   a. Navigate to the User Manager with DBA.Admin privileges and click **New** in User Finder.
   b. In **User Name**, enter a name of your choice.
   c. Enter the client ID from the Oracle Identity Cloud Service confidential application in **Nickname**.
   d. In **Domain Name**, enter the name of the target domain.
   e. Enter a compliant **Password** and the password confirmation.
   f. In **User Role ID**, select **INTEGRATION**.
   g. In **Access Control List** add the list named **Data Replication Service - Rest** with the **Granted** option selected.
   h. Select **Finished**.
3. Obtain the OAuth 2 client credentials from your Oracle Transportation Management Cloud Service instance to provide while creating the data connection. See OAuth 2.
4. In NetSuite Analytics Warehouse, enable **Oracle Transportation Management** on the Enable Features page. See Make Preview Features Available.
5. In NetSuite Analytics Warehouse, create the Oracle Transportation Management Cloud Service data connection:
   a. In NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
   b. On the Data Configuration page, click **Manage Connections**.
   c. On the Manage Connections page, click **Create** and then click **Connection**.
   d. In Create Connection, select **Data Extraction** in **Usage Type** and then select **Oracle Transportation Management** as the connection type.



e. In the dialog for the Oracle Transportation Management Cloud Service connection, enter these details and then click **Save**:

- **Connectivity Type**: Standard.
- **Notification Email**: An email address to receive notifications.
- **Scope**: The scope name defined in your Oracle Transportation Management Cloud Service's Oracle Identity Cloud Service application's configuration.
- **Token URL**: The URL from which to obtain the access token. See Runtime Configuration.
- **URL**: Source application URL (hostname and protocol of the Oracle Transportation Management Cloud Service instance).
- **Client ID**: The unique client identifier generated during OAuth registration process.
- **Client Secret**: The client secret generated during the OAuth registration process (a private key similar to a password that is generated when registering your Oracle Transportation Management Cloud Service's Oracle Identity Cloud Service application).
- Verify that the **Refresh Metadata** toggle is enabled to ensure that the metadata is refreshed when you save the connection. You can later refresh the metadata from the Actions menu on the Manage Connections page, if required.

> **Note:**
>
> You can't create augmentations for Oracle Transportation Management Cloud Service unless you perform a metadata extract.



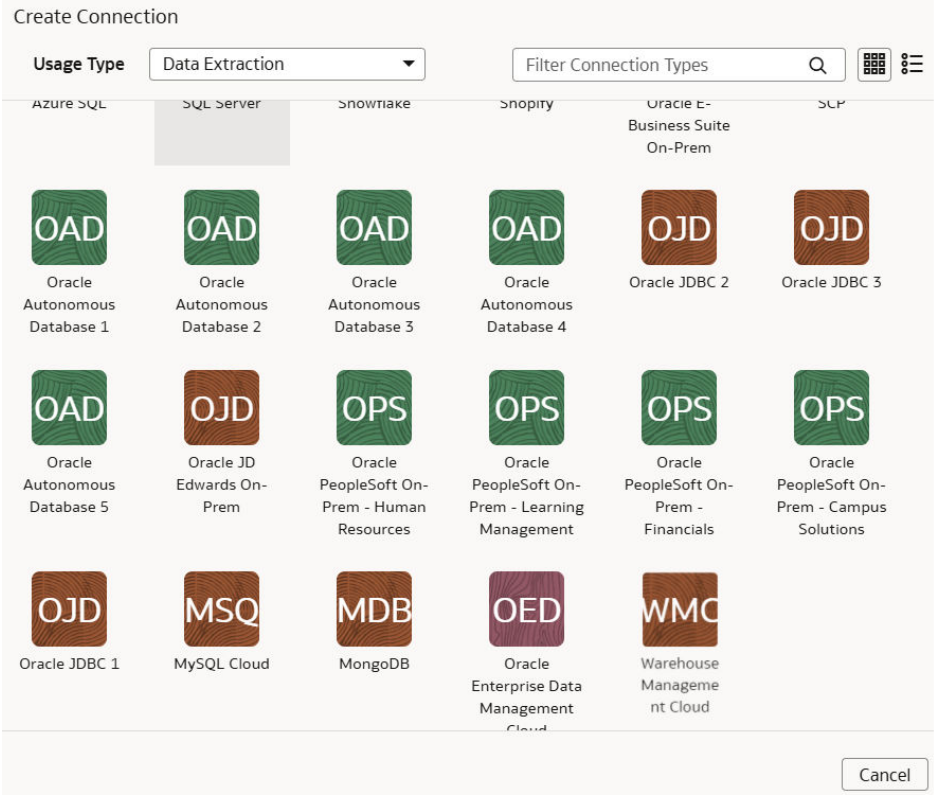
6. On the Manage Connections page, select **Actions** for the Oracle Transportation Management Cloud Service connection and then select **Test Connection**.
7. After the connections are successfully established, navigate to the Data Configuration page, select the connection in **Data Source**, then click the Data Augmentation tile, and create a data augmentation using the Oracle Transportation Management Cloud Service data. Select the applicable Oracle Transportation Management Cloud Service source tables. See Augment Your Data.

Load Data from Oracle Warehouse Management Cloud into Oracle NetSuite Analytics Warehouse (Preview)

As a service administrator, you can use the Oracle NetSuite Analytics Warehouse extract service to acquire data from Oracle Warehouse Management Cloud and use it to create data augmentations.

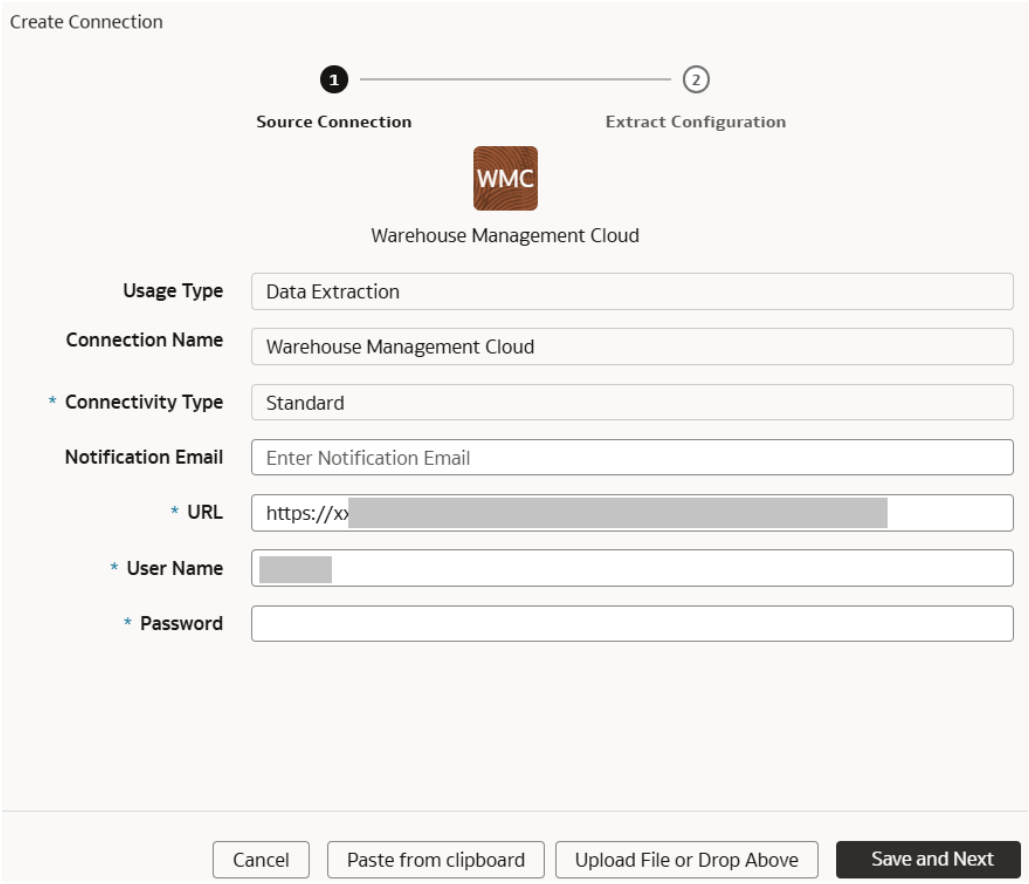
Ensure that **Oracle Warehouse Management Cloud** is enabled on the Enable Features page prior to creating this connection. See Make Preview Features Available.

1. In Oracle NetSuite Analytics Warehouse, create the Oracle Warehouse Management Cloud data connection using these instructions:
   a. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under Application Administration.
   b. On the Data Configuration page, click **Manage Connections**.
   c. On the Manage Connections page, click **Create** and then click **Connection**.
   d. In Create Connection, select **Data Extraction** in **Usage Type**, and then select **Warehouse Management Cloud** as the connection type.



   e. In the dialog for the Oracle Warehouse Management Cloud connection, provide these details in Source Connection and then click **Save and Next**:
      - **Connectivity Type**: Select **Standard**.
      - **Notification Email**: Enter an email address to receive notifications.

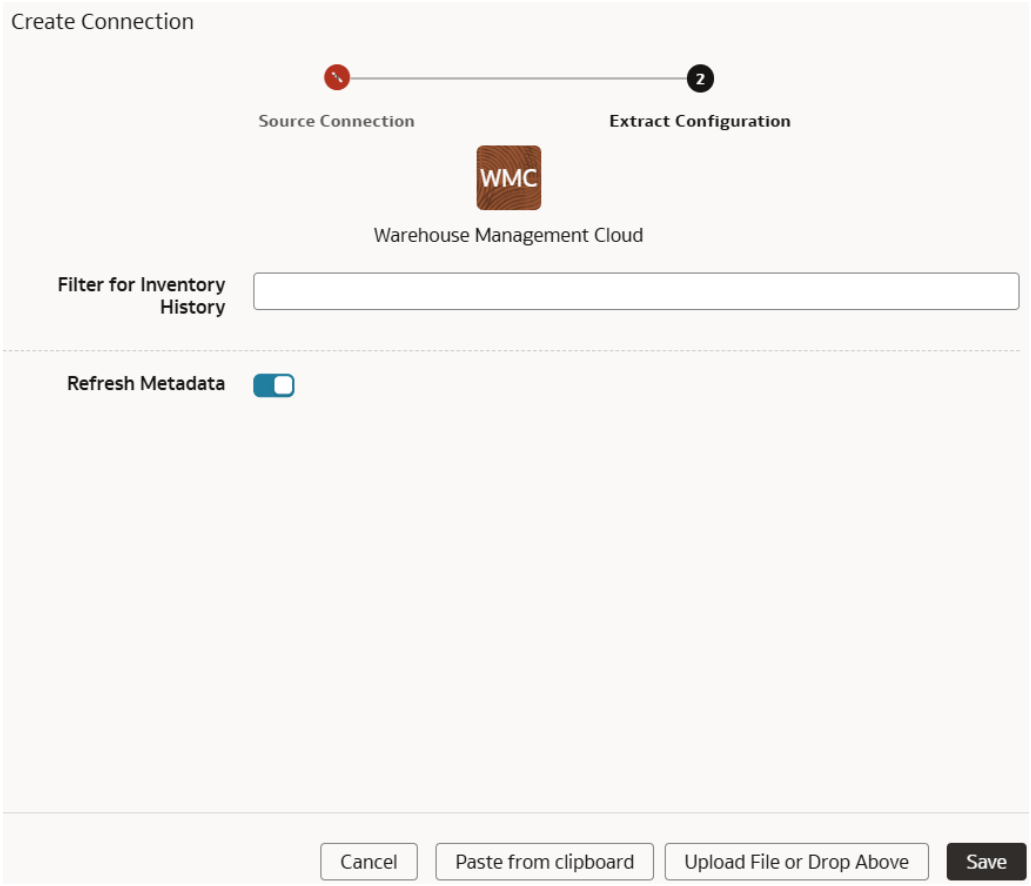
- **URL**: Enter the URL of your Oracle Warehouse Management Cloud instance in this format: `*protocol://domain/environment/app/lgfapi version/lgfapi module/resourcepath*`. For example, `https://xxxxx.wms.ocs.oraclecloud.com/myenv/wms/lgfapi/v10/entity`.
- **User Name** and **Password**: Enter the credentials for your Oracle Warehouse Management Cloud instance.



f. In the dialog for the Oracle Warehouse Management Cloud connection, in Extract Configuration, provide a value in **Filter for Inventory History** using at least one of the following combinations (in addition to any other field) and click **Save**:

```
* company_id_{_}code, facility_id{_}_code, group_nbr
* company_id_{_}code, facility_id{_}_code, history_activity_id,
status_id
* company_id_{_}code, facility_id{_}_code, history_activity_id,
item_code
* company_id_{_}code, facility_id{_}_code, history_activity_id,
item_alternate_code
* company_id_{_}code, facility_id{_}_code, history_activity_id,
container_nbr
```

If you don't enter the filter information, then data extraction for the inventory history datastore won't work. See Oracle Warehouse Management documentation for information about the filter for inventory history.



g. Verify that the **Refresh Metadata** toggle is enabled to ensure that the metadata is refreshed when you save the connection.

   You can later refresh the metadata from the Actions menu on the Manage Connections page, if required.

> **Note:**
>
> You can't create augmentations for Oracle Warehouse Management Cloud unless you perform a metadata extract.

2. On the Manage Connections page, select **Actions** (⋮) for the Oracle Warehouse Management Cloud connection and then select **Test Connection**.
3. After the connections are successfully established, navigate to the Data Configuration page, select the connection in **Data Source**, then click the Data Augmentation tile, and create a data augmentation using the Oracle Warehouse Management Cloud data. Select the applicable Oracle Warehouse Management Cloud source tables. See Augment Your Data.

Schedule Frequent Data Refreshes

You can schedule frequent refreshes of the applicable functional areas, prioritized datasets, and warehouse tables to ensure that the incremental data in these is the latest.

About Scheduling Frequent Data Refreshes

You can refresh specific set of tables currently with plans to support more functional areas and datasets in future releases.

When you refresh certain functional areas, prioritized datasets, and warehouse tables, be sure you understand which tables you can refresh or not because the selective refresh of some tables could lead to functional inconsistencies when combining data from multiple subject areas. This frequent data refresh capability is designed for refresh of base tables that capture the transactional data; it isn't meant for derived datasets that require aggregations, snapshots, or complex transformation logic. Such processing creates data latency that doesn't support high volume of frequent data refresh. For Oracle NetSuite Analytics Warehouse, you can schedule frequent refreshes for functional areas that are visible in the Frequent Data Refresh Schedule tab on the Pipeline Settings page.

If you've enabled the "Prioritized Data Refresh" preview feature and selected datasets for a prioritized incremental refresh, then those specific datasets are available for a frequent data refresh. See Prioritize Datasets for Incremental Refresh (Preview). If you've enabled the "Frequent Refresh Tables" preview feature and saved your selection of the warehouse tables, then the selected tables are available as "Warehouse Tables" for a frequent data refresh. See Schedule Frequent Refreshes of Warehouse Tables (Preview). If you want to select the warehouse tables created by the custom data configurations that were generated prior to release Platform 23.R4, then you must regenerate the applicable custom data configurations for these tables to be visible for selection. From release Platform 23.R4, the warehouse tables created by the custom data configurations are available for a frequent data refresh under the Frequent Refresh tab.

When you select the functional areas for a frequent refresh, you won’t be able to refresh the data pipeline for the applicable functional area using the Refresh Data option on the Data Configuration page. The frequent data refresh process doesn't refresh data from external sources through the data augmentation connectors. Oracle NetSuite Analytics Warehouse processes the data augmentations as part of the incremental loads. If you change a data augmentation after including it in the frequent data refresh schedule, then you must remove that data augmentation and let the next incremental refresh finish. Otherwise, the frequent data refresh might fail. After the incremental refresh is complete, you can add the updated data augmentation back to the frequent data refresh schedule.

For frequent data refreshes, the semantic model won't be updated. The update jobs for semantic model won't run as part of data augmentations, they run for data augmentations only during incremental loads.

Review and consider the following to ensure that frequent data refreshes work as expected:

- Performance of frequent data refreshes depends on the:
  - Size of data.
  - Data change such as what data has changed, and which pipeline gets triggered.
  - Number of extracted records that may result in very different number of published records, for example, 44 extracted records resulted in 1060 published records in 70 minutes and 395 extracted records resulted in 55 published records in 35 minutes.
- The frequent data refresh process doesn’t get executed in the following scenarios:
  - In the 180-minute window before the scheduled start of the daily incremental data refresh.
  - If any release upgrade is in progress.

  - Until the previous frequent data refresh process is completed. You can set 1 hour frequency (maximum), however, in some cases it takes more than 1 hour to complete the refresh; in that case, the next frequent data refresh process starts at the next hour.
- For dataset-level (warehouse tables) frequent data refresh:
  - You must know which exact datasets to refresh.
  - There is a limit of up to 20 datasets for each run.
  - Dependencies aren't automatically incorporated. You must determine the dependencies and include the applicable tables.

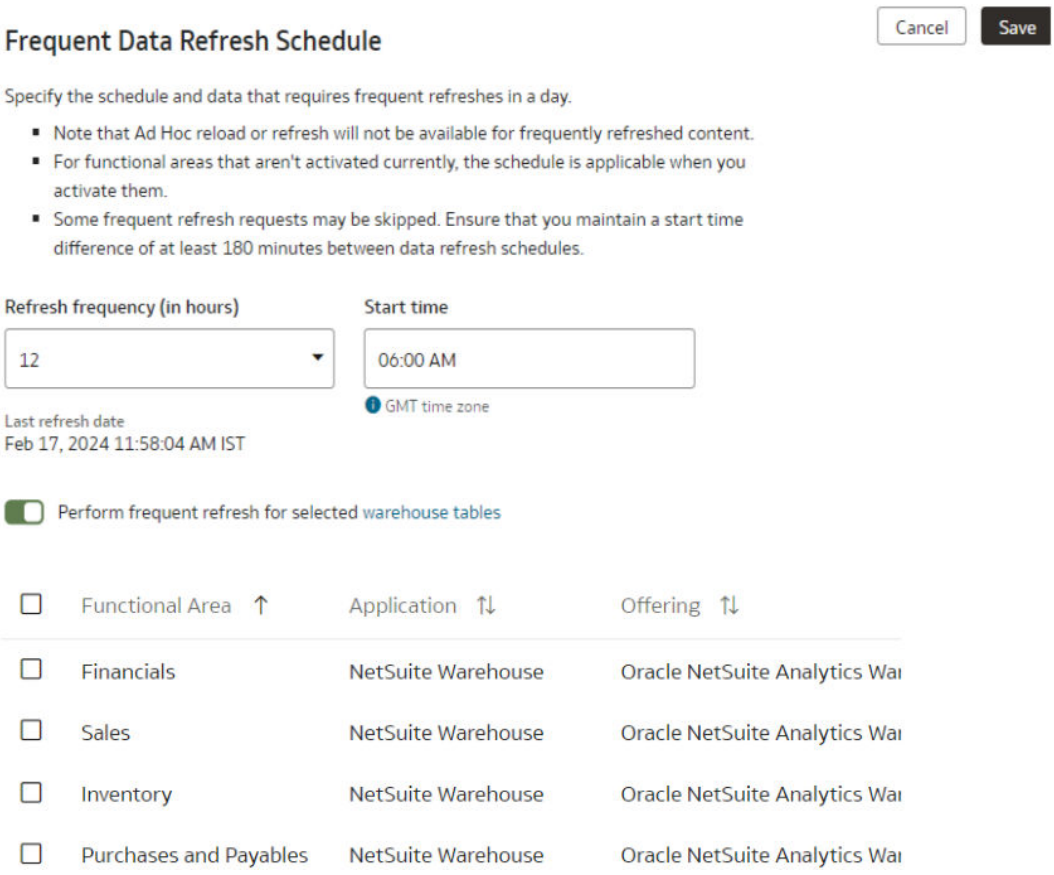
Schedule Frequent Refreshes of Data

You may want to frequently refresh the data in these sample scenarios:

- During the last five days of the month or every quarter, general ledger data changes quickly with updated entries more than once a day. In this scenario, a financial analyst wants to ensure taking more recent and accurate data into the analysis and report generation. Because some of the data pipelines can take several hours to execute from end to end, the frequency of refresh needs to be every 4 hours or 6 times in a 24-hour period.
- In this scenario, a retailer that has strict guidelines for returns or refunds on any purchase orders, and needs to bring in the most recent order data for computation and report generation, especially in the beginning of a given fiscal quarter. To cover for time differences between Europe, East Coast, and West Coast of the U.S., a financial analyst needs to refresh data every 8 hours in 24 hours, or 3 times a day.

Ensure that **Frequent Refresh** is enabled in **Pipeline Features** under the Generally Available Features tab on the Enable Features page prior to scheduling frequent data refreshes. See Enable Generally Available Features.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
3. On the Data Configuration page, under **Configurations**, click **Pipeline Settings**.
4. On the Pipeline Settings page, click **Frequent Data Refresh Schedule**.



5. In the Frequent Data Refresh Schedule tab, in **Refresh Frequency (in hours)**, select the number of hours after which you want the data refresh to happen.
6. Enter a start time in **Start time**.
7. In **Last refresh date**, view the date when the data was last refreshed.
8. If you've enabled the "Frequent Refresh Tables" preview feature and saved your selection of the warehouse tables, then select the **Perform frequent refresh for selected warehouse tables** toggle. Click the "warehouse tables" hyperlink to view or update your selection of the warehouse tables in the Frequent Refresh Tables tab.
9. Select the check boxes for the functional areas with the data you want refreshed at the specified interval.
10. Click **Save**.

Schedule Periodic Full Reload of Functional Area Data

You can schedule a periodic full reload of the applicable functional areas to ensure that all the data in these functional areas is completely reloaded.

For Oracle NetSuite Analytics Warehouse, you can schedule a periodic full reload for functional areas such as Financials and Sales. When you select the functional areas for full reload, ensure that you select the dependent functional areas as well.

> **WARNING:**
>
> When you schedule a full reload, the incremental loads take a longer time and depending on the initial extract date, the incremental data load may take days to complete the extraction.

You may want to fully reload the data for a functional area in these sample scenarios:

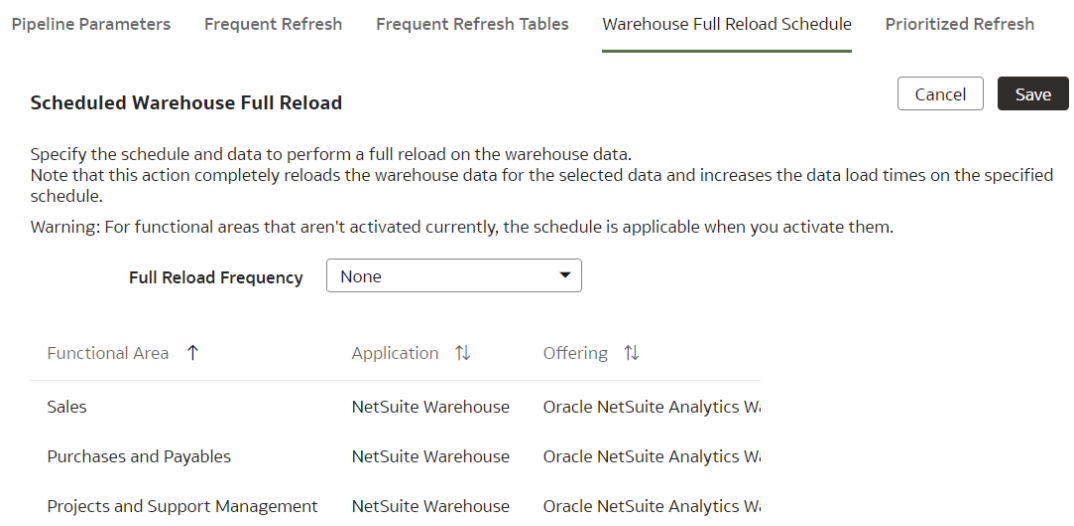
- The Finance department wants to do full reload of the general ledger (GL) data at the beginning of the month to make sure all data is fresh and clean for the monthly GL entries and the analytics and report generation.
- At the beginning of every quarter, all data related to Purchase Orders is updated with returns, refunds, and any adjustments. In this scenario, revenue recognition needs to take all the latest Purchase Order data to compute all numbers and create accurate reports. Hence, a full reload of the order data is required on the 1st of every fiscal quarter that can be different from the calendar quarter.

You can check the refresh summary in two ways:

- View details of the requests on the Request History page.
- View the DW_WH_REFRESH_SUMMARY data warehouse table.

Ensure that **Scheduled Full Data Reload** is enabled in **Pipeline Features** under the Generally Available Features tab on the Enable Features page prior to scheduling periodic full reload. See Enable Generally Available Features.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
3. On the Data Configuration page, under **Configurations**, click **Pipeline Settings**.
4. On the Pipeline Settings page, click **Warehouse Full Relaod Schedule**.



5. In the Warehouse Full Reload Schedule tab, in **Full Reload Frequency**, select the frequency type such as weekly, monthly, or quarterly.

6. Based on the selected frequency type, select the specific day of the week, month, or the first month of the quarter.
7. Select the check boxes for the functional areas whose data you want reloaded fully at the specified period.
8. Click **Save**.

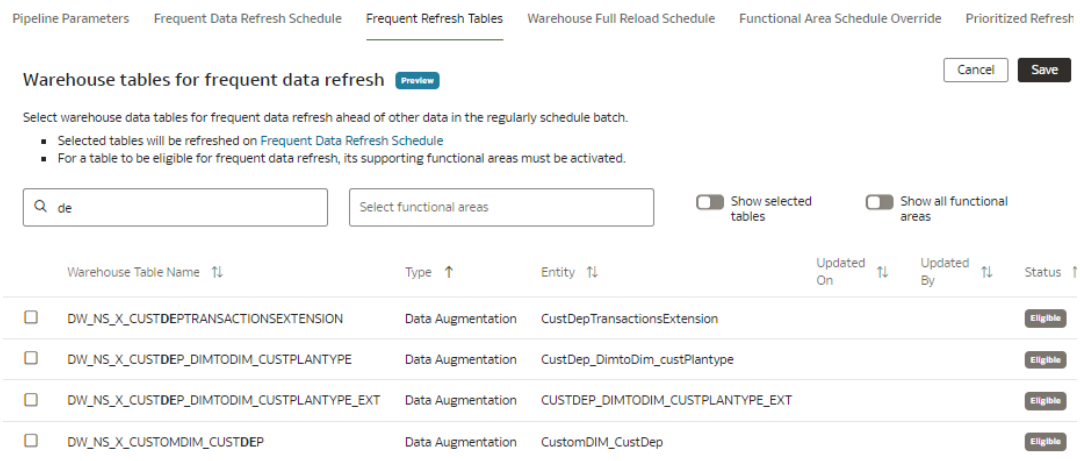
Schedule Frequent Refreshes of Warehouse Tables (Preview)

You can select warehouse data tables that are part of functional areas, descriptive flexfields custom configurations, and data augmentations for frequent data refresh ahead of other data in the regularly scheduled batch.

For a table to be eligible for a frequent data refresh, ensure that its supporting functional areas, data augmentations, and descriptive flexfields custom configurations are activated. The custom data configurations activated prior to release 23.R4 won't be available automatically. You must regenerate and redeploy them to schedule for frequent refresh. After you select the warehouse tables for frequent data refresh and save your selection, the selected tables are available as "Warehouse Tables" under the Frequent Refresh tab. You can set the frequent refresh schedule for these tables in the Frequent Refresh tab. See Schedule Frequent Refreshes of Data.

Ensure that **Frequent Refresh Tables** is enabled in **Pipeline Features** under the Preview Features tab on the Enable Features page prior to scheduling frequent refreshes of the tables. See Make Preview Features Available.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
3. On the Data Configuration page, under **Configurations**, click **Pipeline Settings**.
4. On the Pipeline Settings page, click **Frequent Refresh Tables**.



5. In the Frequent Refresh Tables tab, select the check boxes for the applicable warehouse tables and click **Save**.

Override Data Pipeline Schedules for Functional Areas (Preview)

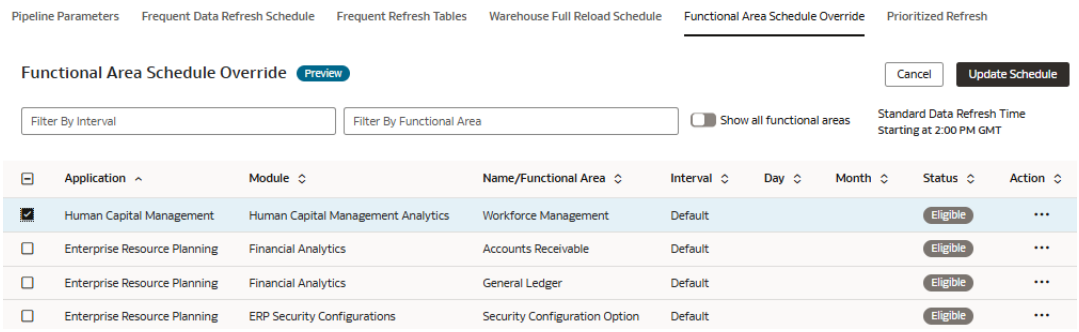
You can remove a functional area from the standard incremental schedule and set a separate schedule with different frequency of run to prioritise your data pipelines and run timings.

You can select one or more functional areas and schedule to run incremental at a different frequency than the standard incremental job. If the incremental refresh is scheduled for Daily, then you can choose one or more functional areas to schedule them to run Weekly, Monthly, Quarterly, or Yearly. If the incremental refresh is scheduled for Weekly, then you can choose one or more functional areas to schedule them to run Monthly, Quarterly, or Yearly.

If the incremental refresh is weekly, then it runs on a certain day of the week. Thereafter, if you do an override for Monthly on a certain day of the month, then you can't piggyback on incremental since weekly incremental (certain day of the week, let's say Monday) may not run on the desired Day of the Month (let's say 1st). In this scenario, you can create schedules to override different modules on different days of the week and override different modules on the same day of the week.

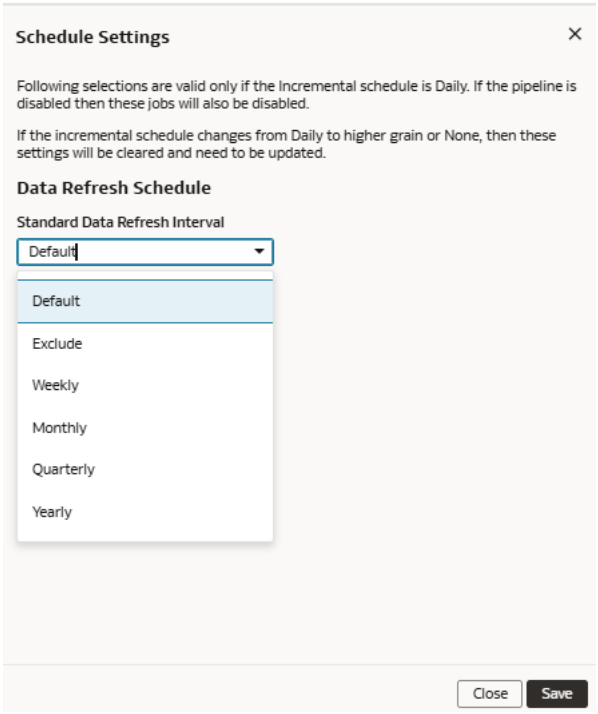
Ensure that **Functional Area Schedule Override** is enabled in **Pipeline Features** under the Preview Features tab on the Enable Features page . See Make Preview Features Available.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
3. On the Data Configuration page, under **Configurations**, click **Pipeline Settings**.
4. On the Pipeline Settings page, click **Functional Area Schedule Override**.



5. In the Functional Area Schedule Override tab, select the applicable functional areas and click **Update Schedule**.

   To set the schedule for an individual functional area, click the **Action** menu for the applicable function area and click **Edit Schedule**.
6. In Schedule Settings, specify the data refresh schedule by selecting a value in the **Standard Data Refresh Interval** field and the dependent values for **Start Day** and **Month** when applicable.



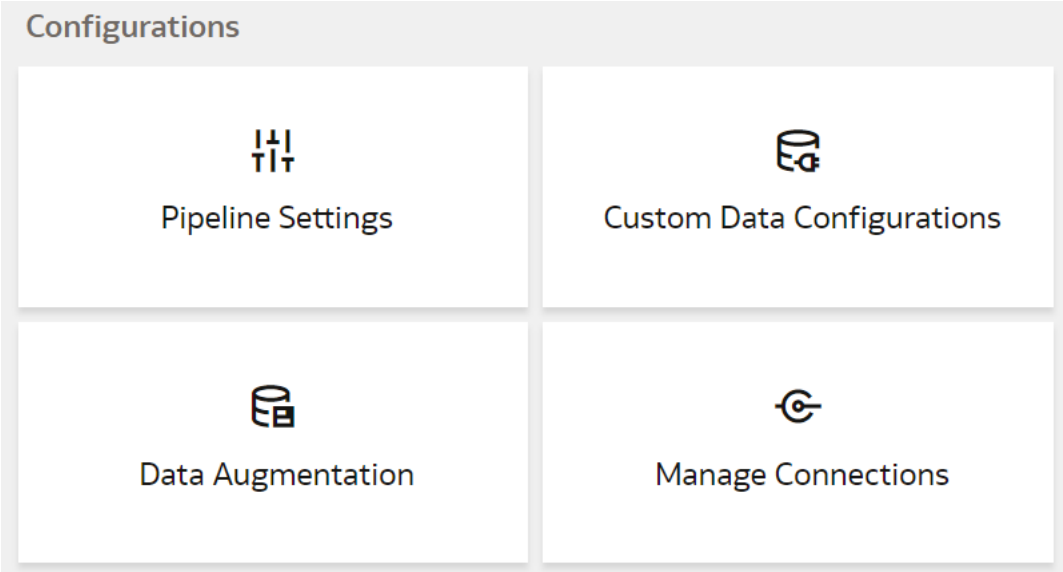
7. Click **Save**.

Prioritize Datasets for Incremental Refresh (Preview)

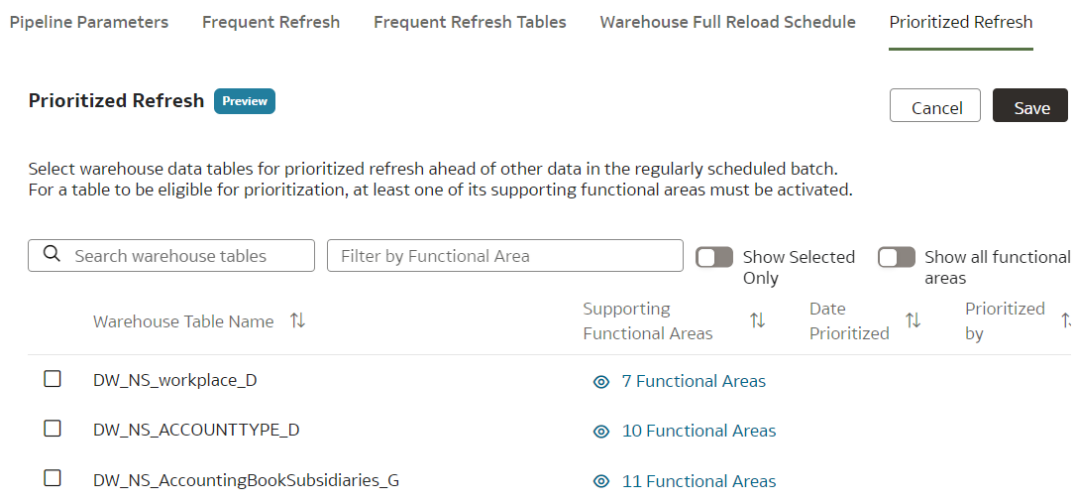
As a functional administrator, you can select datasets to prioritize for incremental refresh over others. For example, at the end of the quarter, you want to prioritize some of the datasets that have bookings data over other datasets.

You can search for and choose datasets based on the functional areas. The incremental refresh process automatically selects the dependent datasets. For example, if you select DW_GL_JOURNAL_CF, then the incremental refresh process automatically pulls in DW_GL_JOURNAL_F. The incremental refresh process runs the priority dataset refresh first from end-to-end, followed by the regular incremental refresh. The objects processed in the priority dataset refresh are reprocessed in the regular incremental refresh. The prioritized datasets are automatically available for frequent data refresh. See Schedule Frequent Refreshes of Data.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
3. On the Data Configuration page, under **Configurations**, click **Pipeline Settings**.



4. On the Pipeline Settings page, click the Prioritized Data Refresh tab.



5. In the Prioritized Data Refresh section, select the datasets that you want for the priority refresh.
6. Click **Save**.

Reset and Reload the Data Source

As a functional administrator, you can reset and reload the data source for several reasons.

For example:

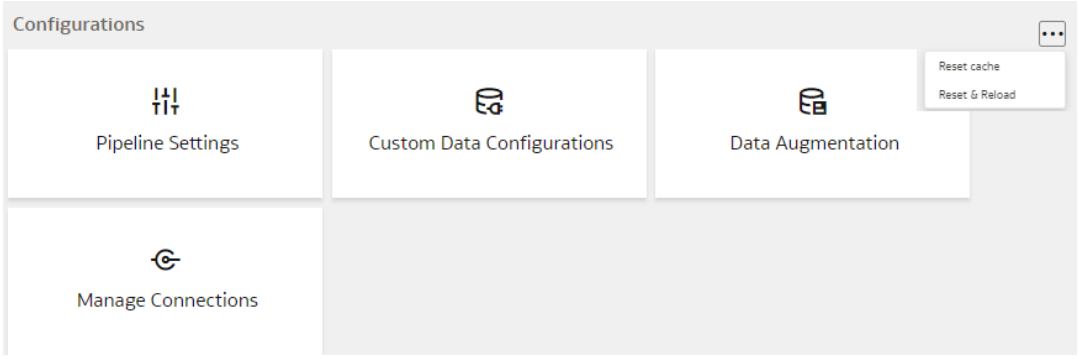
- If you need to change the data source for some reason, then you may want to reset the present data source back to the original state.
- If there is any data issue with the source system and you need to start the entire pipeline all over again.

- Initially if you set up the connection and the pipeline to test some things and then want to start from the beginning again.
- Whenever you perform a Production to Test (P2T) refresh, you must complete a full reload of the source to avoid data issues.

You can reset and reload data for all the activated functional areas, augmentations, and custom data configurations. This request may take a long time as it requires all data to be discarded and fetched again. Other requests like the incremental and full load won't start until this request completes.

Ensure that **Reset and Reload the Data Source** is enabled in **Pipeline Features** under the Generally Available Features tab on the Enable Features page prior to resetting and reloading the data source. See Enable Generally Available Features.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration** under **Application Administration**.
3. On the Data Configuration page, click **Reset & Reload**.



4. Review the warning message, provide the details, and then click **Submit**.

Extend Data with Custom Data Configurations

As a functional administrator, you can use the available custom data configurations to extend your data for detailed and varied analytics.

Topics:

About Extending Data with Custom Data Configurations

Custom data configurations are prebuilt templates that enable you to create custom functional areas with the data that you bring into the Autonomous Data Warehouse. You later use these

custom functional areas to augment the data from the source or enrich the data available in the Autonomous Data Warehouse.

When you start using a custom data configuration for the first time and deploy it, Oracle NetSuite Analytics Warehouse assigns the custom data configuration a version number. You can then edit the deployed version, make changes, and save it. This saved version then gets a new version number. While you're changing the definition, the deployed version continues to fetch new data. An example of the currently available custom data configuration is Netsuite Account Analysis.

About Extending Data with Netsuite Account Analysis

NetSuite Account Analysis is a custom data configuration that enables you to build custom content by choosing data across 42 transaction types.

It also supports posting custom transactions, custom attribute mapper based custom attributes, and prebuilt subject areas. This feature supports granularity at the transaction line level. Once custom data configuration is deployed, the created content manifests in the form of a subject area. This feature eliminates the need for users to go to raw table and link them using SQL to build across transaction type content for those 42 transaction type.

Prerequisites for Setting up Netsuite Account Analysis

Ensure these prerequisites are met:

- Prior to configuring the Netsuite account analysis template, ensure that you've activated at least one functional area such as Sales. See Activate a Data Pipeline for a Functional Area.
- If you want to use the custom transactions and custom attributes while creating the account analysis subject area, then contact Oracle Support to have the custom transactions and custom attributes enabled in your instance.
- If you want to use custom attributes in Netsuite account analysis, then ensure that the Custom Attribute Mapper (CAM) entities are published before any other action.

Enable Netsuite Account Analysis (Preview)

Prior to configuring the Netsuite account analysis template to build the account analysis reports in Oracle NetSuite Analytics Warehouse, you must enable it.

1. Sign in to your service as a functional administrator.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Enable Features**, and then click **Preview Features**.
3. Under the Preview Features tab, select the toggle to enable **Netsuite Account Analysis**.

Set up Netsuite Account Analysis

Set up the Netsuite Account Analysis subject area to build the account analysis reports in Oracle NetSuite Analytics Warehouse.

While configuring the custom account analysis subject area, ensure that you note the value in the Usage column for each of the available columns on the Netsuite Account Analysis page:

- Certain columns are mandatory for the account analysis report, hence these columns are selected by default and you can't deselect them.

- Certain columns are recommended and selected by default but you can deselect them based on your requirements for the account analysis reports.
- Certain columns are optional and you can select them based on your requirements for the account analysis reports.
- Certain columns are custom attributes and custom transactions-related. They're available for selection if you've completed the prerequisite task. See Prerequisites for Setting up Netsuite Account Analysis. After completing the prerequisite task, you must select the **Include custom transaction applicable for Account Analysis** check box to have the columns from custom transactions available for selection.

After creating the custom account analysis subject area, you can create the account analysis reports based on your business requirements.

1. Sign in to your service as a functional administrator.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Data Configuration**, and then on the Data Configuration page, click **Custom Data Configurations**.
3. On the Custom Data Configurations page, click **Create** and then select **Netsuite Account Analysis**.
4. On the Netsuite Account Analysis page, select the data elements that you want to include in the custom account analysis template, and then click **Save and Exit**.
5. On the Custom Data Configurations page, click **Actions** for the Netsuite Account Analysis that you created, and then click **Generate**.

   You can monitor the status of the Generate job on the Request History page.
6. On the Custom Data Configurations page, when you see the status of the Netsuite Account Analysis that you created as **Generated**, then click **Actions** and select **Deploy**.

After you've successfully deployed the Netsuite Account Analysis subject area, on the Custom Data Configurations page, you can perform these actions by clicking **Actions** for the Netsuite Account Analysis that you deployed:

- Undeploy
- Reload Data
- Edit
- Deactivate
- Refresh Data
- Delete

3

Manage Users, Groups, and Application Roles

As the service administrator or security administrator, you manage users and their access to subject areas and objects.

Topics:

About Managing Users, Groups, and Application Roles

As the service administrator or security administrator, one of your initial tasks is to ensure that users have appropriate access to use Oracle NetSuite Analytics Warehouse.

Users need access to objects and data. Access to objects include subject areas or elements of subject areas such as folders and attributes, key metrics, workbooks, and the legacy Oracle BI Enterprise Edition dashboards and answers. You grant access to the users by assigning groups to them. The groups inherit the permissions from the application roles including the licensed roles (from release Platform 23.R4) mapped to them.You set up your security components on the production environment.

The enhanced security capabilities available from release Platform 23.R4 enable you to use the licensed application roles corresponding to the existing licensed groups made available in this release onwards. You can assign the licensed application roles to your custom groups. These licensed application roles are mapped by default to the existing ready-to-use licensed groups and are associated with the system roles. When you assign these ready-to-use licensed groups and custom groups to users, then the applicable users get the system privileges such as consumer, author, and administrator. You can continue to use the existing security capabilities till Oracle NetSuite Analytics Warehouse automatically applies them as part of the release 24.R2. These changes are mandatory and you can schedule the update sooner to enhance the security of your application.

About Users

Users accessing Oracle NetSuite Analytics Warehouse must exist in Oracle Identity Cloud Service.

You can create the users or synchronize them with the Oracle Identity Cloud Service instance associated with your Oracle NetSuite Analytics Warehouse instance from different sources:

- You can synchronize the Netsuite users with the Oracle Identity Cloud Service instance using the Configuration page for Oracle NetSuite Analytics Warehouse in NetSuite.
- You can manually create users in the Oracle Identity Cloud Service instance using the Oracle NetSuite Analytics Warehouse user interface.

- You can synchronize the users from other 3rd-party systems with the Oracle Identity Cloud Service instance.

Users gain their access to Oracle NetSuite Analytics Warehouse based on the NetSuite Analytics Warehouse-specific system groups assigned to them. They gain access to different functionality, objects, and data in Oracle NetSuite Analytics Warehouse based on the job-specific groups assigned to them.

You can assign the predefined system groups, groups available in the Oracle Identity Cloud Service instance associated with your Oracle NetSuite Analytics Warehouse instance, and custom groups that you create in Oracle NetSuite Analytics Warehouse. See Associate Users and Groups.

About Groups

Oracle NetSuite Analytics Warehouse uses groups to provide users access to subject areas, objects, and data.

Oracle NetSuite Analytics Warehouse uses the following three types of groups:

- System groups created in Oracle Identity Cloud Service specifically for Oracle NetSuite Analytics Warehouse. These system groups are associated with system roles that provide a set of privileges to the users to perform system tasks after signing into Oracle NetSuite Analytics Warehouse, such as administering system settings, performing functional setup, managing security, and modeling data.
- Other groups that are generic groups created in Oracle Identity Cloud Service not specifically for Oracle NetSuite Analytics Warehouse, such as IDCS_Administrators and All_Tenant_Users.

System Groups

Oracle NetSuite Analytics Warehouse creates the system groups in Oracle Identity Cloud Service while provisioning your Oracle NetSuite Analytics Warehouse instance.

System groups are associated with system roles that provide a set of privileges to users. The enhanced security capabilities available from release Platform 23.R4 provide licensed application roles corresponding to the existing licensed groups and are mapped by default to the existing ready-to-use system or licensed groups. The system roles or the licenses application roles (from release Platform 23.R4) serve two purposes:

- Authenticate a user for Oracle NetSuite Analytics Warehouse.
- License a user to use Oracle NetSuite Analytics Warehouse based on the system group they are assigned.

See System Roles and Licensed Roles.

You must add the users to the corresponding system groups based on the tasks they perform in Oracle NetSuite Analytics Warehouse. See Assign Users to a Group and Assign Groups to Users. Use the Users or Groups tabs on the Security page to add users to these system groups:

System Group Code	System Group Name	Description	Associated System Role*	Associated Licensed Application Role from release Platform 23.R4 Onwards
NAW_Licensed_Authors	NAW Licensed Authors	NetSuite Analytics Warehouse Licensed Users	Author	NAW Licensed Authors Role
NAW_Licensed_Users	NAW Licensed Users	NetSuite Analytics Warehouse Licensed Users	Author	NAW Licensed Authors Role
NAW_Licensed_Viewers	NAW Licensed Viewers	NetSuite Analytics Warehouse Licensed Viewers	Consumer	NAW Licensed Viewers Role
NAW_Service_Admin	NAW Service Admin	NetSuite Analytics Warehouse Service Administrators	ServiceAdmin	NAW Service Admin Role

Other Groups

The Other Groups category refers to groups created in Oracle Identity Cloud Service for purposes such as administrating Oracle Cloud Infrastructure and Oracle Identity Cloud Service.

These groups aren't necessarily Oracle NetSuite Analytics Warehouse-specific but you can use them in Oracle NetSuite Analytics Warehouse. Examples of this category are the "IDCS_Administrators" and "All_Tenant_Users" groups.

About Application Roles

Application roles in Oracle NetSuite Analytics Warehouse consist of duty roles.

Duty roles define the duties of a job as an entitlement to perform a particular action; for example, access to a Sales functional area-related subject areas.

Duty Roles

The predefined duty roles to secure the predefined subject areas and the predefined front-end objects are:

Duty Role Name	Duty Role Description	Functional Area	Gets access to Subject Area Display Name OR Associated Role
NetSuite Analytics Warehouse Sales Duty	Object security role to control presentation catalog access to Sales functional area.	Sales	All subject areas under the Sales functional area.
NetSuite Analytics Warehouse Purchases Duty	Object security role to control presentation catalog access to Purchases functional area.	Purchases	All subject areas under the Purchases functional area.

Duty Role Name	Duty Role Description	Functional Area	Gets access to Subject Area Display Name OR Associated Role
NetSuite Analytics Warehouse Inventory Duty	Object security role to control presentation catalog access to Inventory functional area.	Inventory	All subject areas under the Inventory functional area.
Data Warehouse Refresh and Usage Tracking Analysis Duty	Object security role to control presentation catalog access to Warehouse Refresh and Usage Tracking subject areas.	Not applicable	Warehouse Refresh and Usage Tracking subject areas.
NetSuite Analytics Warehouse Financials Duty	Object security role to control presentation catalog access to Financials functional area.	Financials	All subject areas under the Financials functional area.
NetSuite Analytics Warehouse Manufacturing Duty	Object security role to control presentation catalog access to Manufacturing functional area.	Manufacturing	All subject areas under the Manufacturing functional area.
NetSuite Analytics Warehouse Payroll Duty	Object security role to control presentation catalog access to Payroll functional area.	Payroll	All subject areas under the Payroll functional area.

Licensed Roles

The licensed application roles corresponding to the existing licensed groups made available in release Platform 23.R4 onwards are as follows.

The licensed application roles are by default associated with the applicable system roles described in System Roles.

Licensed Role	Associated System Role	Mapped to Licensed Group
NAW Service Admin Role	Service Administrator	NAW Service Admin
NAW Licensed Authors Role	Author	NAW Licensed Author
NAW Licensed Viewers Role	Consumer	NAW Licensed Viewer

System Roles

The system roles for Oracle NetSuite Analytics Warehouse available in Oracle Identity Cloud Service through provisioning of Oracle NetSuite Analytics Warehouse are:

Role Name	Role Description	Purpose	Permissions
Service Administrator	Oracle NetSuite Analytics Warehouse service administrator	Customer facing (Snapshots, Connections, System Settings) administrator access to Oracle NetSuite Analytics Warehouse.	• Can't create snapshots or modify the data model file (RPD) • Can access the Data Pipeline user interface • Can access the Data Security user interface • Has no access to the Job Monitoring console • Can access the Console menu • Can access the user and group administration pages • Can access the Semantic Model Extensions user interface • Has read-only access to the ready-to-use Oracle Analytics Cloud objects (visualization projects, dashboards, and analyses) • Requests from Oracle NetSuite Analytics Warehouse to Oracle Analytics Cloud are routed through the Service Administrator user • Can create, update, and delete the Oracle Analytics Cloud content • Has read-only access to the ready-to-use KPIs • Can create, update, and delete KPIs • Can create, update, and delete decks and cards • Can share decks and cards • Can create Oracle Analytics Publisher reports

Role Name	Role Description	Purpose	Permissions
			• Has no access to data modeling • Has access to create Oracle Analytics Cloud connections to other non-Oracle Applications sources, such as Excel files and Google drive • Has access to create Oracle Analytics Cloud datasets

Role Name	Role Description	Purpose	Permissions
Author	Oracle NetSuite Analytics Warehouse author	Create and edit KPIs, cards, decks, visualization projects, reports, and dashboards.	• Has no access to the Data Pipeline user interface • Has no access to the Data Security user interface • Has no access to the Job Monitoring console • Has no access to the Console menu • Has no access to user and role administration • Has no access to the Semantic Model Extensions user interface • Has read-only access to the ready-to-use Oracle Analytics Cloud objects (visualization projects, dashboards, and analyses); if you need a change, then create a copy using "Save As" • Has read-only access to the ready-to-use KPIs • Can edit the custom Oracle Analytics Cloud objects (visualization projects, dashboards, and analyses) • Can edit the custom KPIs • Can change the filter values for existing visualization projects • Can add filters for existing visualization projects • Can create and edit Oracle Analytics Cloud content, KPIs, decks, and cards • Can delete custom KPIs, decks, and cards • Can consume KPIs, cards, and decks

Role Name	Role Description	Purpose	Permissions
			created by other users on which they have access permissions • Can share decks and cards • Can create Oracle Analytics Publisher reports • Has no access to data modeling • Has no access to create Oracle Analytics Cloud connections • Has access to create Oracle Analytics Cloud datasets

Role Name	Role Description	Purpose	Permissions
Consumer	Oracle NetSuite Analytics Warehouse consumer	Read access to Oracle Analytics Cloud content and can create cards and decks.	• Has no access to the Data Pipeline user interface • Has no access to the Data Security user interface • Has no access to the Job Monitoring console • Has no access to the Console menu • Has no access to user and role administration • Has no access to the Semantic Model Extensions user interface • Has read-only access to the ready-to-use Oracle Analytics Cloud objects (visualization projects, dashboards, and analyses) • Has read-only access to the ready-to-use KPIs • Has read-only access to the custom Oracle Analytics Cloud objects (visualization projects, dashboards, and analyses) • Has read-only access to the custom KPIs • Can change the filter values for the existing visualization projects • Can't add filters for the existing visualization projects • Can't create any Oracle Analytics Cloud content • Can't create any KPIs

Role Name	Role Description	Purpose	Permissions
			• Can create, update, and delete decks and cards • Can share decks and cards • Has read-only access to Oracle Analytics Publisher • Has no access to data modeling • Has no access to create Oracle Analytics Cloud connections • Has access to create Oracle Analytics Cloud datasets

About Data Access through Security Assignments

You grant the data security assignments at the user-level.

Data security assignments apply data filters to display only the data corresponding to the security assignment values assigned to the users. For example, John Smith and Marie Pierce are both Accounts Payable Manager in an organization, but John Smith needs to see only the US business unit-specific data and Marie needs to see only the UK business unit-specific data. Even though both have the same functional role, their data security assignments differ. John is assigned all the US business units and Marie is assigned all the UK business units only.

You ensure data-level security with a combination of data roles, security context, and security assignments assigned to the user. Oracle NetSuite Analytics Warehouse maps a security context 1:1 onto a data role. You grant the data security assignments within a security context. The user must have the data role through the group assigned to them in order to have access to the security context and its corresponding list of values to pick from. You assign a user one or more job-specific groups. The groups have data roles mapped to them, and when querying data, the semantic layer applies the data filters.

Manage Users

As a service or security administrator, you must ensure that you add users with applicable permissions to work with Oracle NetSuite Analytics Warehouse.

Topics:

- Assign Groups to Users
- Remove Groups Assigned to a User

About Password Policy for Users Created in Oracle NetSuite Analytics Warehouse

The password policy for the users created in Oracle NetSuite Analytics Warehouse is as follows:

- The enforced password requirements are:
  - Passwords must not contain the user's first name, last name, or username.
  - Oracle NetSuite Analytics Warehouse remembers the 4 previous passwords.
  - Password length (minimum) must be 12.
  - Password length (maximum) must not cross 40.
- The complexity required in the password is:
  - Minimum 1 numeric character.
  - Minimum 1 lowercase character.
  - Minimum 1 uppercase character.
- The account is locked after 5 unsuccessful attempts for a duration of 30 minutes.
- The passwords don’t expire.
- Any user can reset passwords and a user can reset their own passwords.

Create Users in Oracle Identity Cloud Service

Create users in the Oracle Identity Cloud Service instance associated with your Oracle NetSuite Analytics Warehouse instance.

See Create User Accounts in *Administering Oracle Identity Cloud Service*.

Create Users in Oracle NetSuite Analytics Warehouse

Manually create users in the Oracle Identity Cloud Service instance using the Oracle NetSuite Analytics Warehouse user interface.

When you add a user using the Oracle NetSuite Analytics Warehouse user interface, the user is available in the Oracle Identity Cloud Service instance associated with your Oracle NetSuite Analytics Warehouse instance. You later assign the applicable licensed groups to the user that enable the user to perform certain tasks in Oracle NetSuite Analytics Warehouse.

See System Groups.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Security** under **Service Administration**.
3. On the Security page, click the Users tab, and then click **Add User**.
4. In Add User, click **Create a New User**.
5. In Add User, provide the user details and click **Next**.

6. Select the check box for each of the licensed groups that you want to assign to the user and click **Finish**.

License the Users to Access Oracle NetSuite Analytics Warehouse

You must assign at least one system group to users to enable them to perform relevant tasks in Oracle NetSuite Analytics Warehouse. System groups provide a set of privileges to perform tasks in Oracle NetSuite Analytics Warehouse.

See System Groups.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Security** under **Service Administration**.
3. On the Security page, click the Users tab, and then click **Add User**.
4. In Add User, search for a user and select the user or select a user from the displayed list of users.
5. Click **Next**.
6. Select the check box for each of the licensed groups that you want to assign to the user and click **Finish**.

Update the User Details

You can update the user details such as first name, last name, and email but the user name is non-editable after specifying it initially.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Security** under **Service Administration**.
3. On the Security page, select a user from the users listed under the Users tab or use the Search option and then click **User Profile**.
4. In User Details, update the applicable information and click **Save**.

Remove a User's Access to Oracle NetSuite Analytics Warehouse

Remove a user's access to Oracle NetSuite Analytics Warehouse by removing their assignment to a system group.

A user assigned to a system group consumes an Oracle NetSuite Analytics Warehouse license. When a user no longer needs to consume the Oracle NetSuite Analytics Warehouse license according to their entitlement, you can revoke the user's assignment to the applicable system groups. The user won't be able to access Oracle NetSuite Analytics Warehouse as the user's access to Oracle NetSuite Analytics Warehouse depends on the system group assigned to them. When you remove all system groups from the user, then the user can no longer be authenticated to Oracle NetSuite Analytics Warehouse.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Security** under **Service Administration**.
3. On the Security page, select a user from the displayed users or use the Search option to search for and select a user.

4. Click **Remove User**.

Copy Data Security Assignments

As a service or security administrator, you can copy data security assignments from one user to another user.

Copying bulk assignments could take some time to process. Monitor the Activity tab on the Security page.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Security** under **Service Administration**.
3. On the Security page, select a user from the users listed under the Users tab or search and select a user.
4. Click **Copy Assignments**.
5. In Copy Security Assignments From Another User:
   a. Under Copy From, search for and select a user to copy access from.
   b. Under Copy Security Access, you see the context-specific security assignments that would be copied.
6. Click **Copy**.

Assign Groups to Users

You can assign one or more users to one or more groups.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Security** under **Service Administration**.
3. On the Security page, select a user from the displayed users or use the Search option to search for and select a user.
4. In the user details region, click **Assign Groups**.
5. In Assign Groups, search for a group or select from the list of groups displayed in this dialog.
6. Select the check box for one or more groups and click **Assign**.

Remove Groups Assigned to a User

Remove groups assigned to a user if the user no longer requires the authorization to access Oracle NetSuite Analytics Warehouse, specific permissions for tasks, and functional access.

After you remove a group from a user, the applicable user may need to logout for the change to take affect. There could be a gap of around 30 seconds for the refresh to happen.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Security** under **Service Administration**.
3. On the Security page, select a user from the displayed users or use the Search option to search for and select a user.

You see the groups assigned to the user under the Groups tab in the user details region.

4. Under the Groups tab, select the check box for one or more groups and click **Remove Group**.

Manage Groups

You must ensure to map application roles to groups and add user memberships to groups. This enables users to access the applicable objects in Oracle NetSuite Analytics Warehouse and perform various tasks.

Topics:

Create a Group

As a security administrator, you can create custom groups to meet your business requirements.

You can create them manually in the Oracle Identity Cloud Service instance associated with your Oracle NetSuite Analytics Warehouse instance using the Security pages in Oracle NetSuite Analytics Warehouse.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Security** under **Service Administration**.
3. On the Security page, click the Groups tab, and then click **New Group**.
4. In Create a New Group , enter a group name and description.
5. Click **Save**.

Remove a Group

You can remove only the custom groups. When you remove a custom group, Oracle NetSuite Analytics Warehouse removes the associated mappings of the application roles.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Security** under **Service Administration**.
3. On the Security page, click the Groups tab.
4. In the Groups region, search for a group and select it or select a group from the displayed list of groups.

5. Click **Remove Group**.

Add Application Roles to a Group

As a security administrator, you can map the application roles available for Oracle NetSuite Analytics Warehouse with the predefined and custom groups. This enables the groups to inherit the security setup at each application role level.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Security** under **Service Administration**.
3. On the Security page, click the Groups tab.
4. In the Groups region, search for a group and select it or select a group from the displayed list of groups.
5. In the group details region, click the Application Roles tab, and then click **Add Mapping**.
6. In Add Application Role Mappings, search for an application role and select it or select from the displayed list of application roles.
7. Click **Save**.

Copy Application Roles to a Group

As a security administrator, you can copy the application roles available from an existing group to another group.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Security** under **Service Administration**.
3. On the Security page, click the **Groups** tab.
4. In the Groups list, select the group to which you want to apply the application roles
5. On the Groups tab, click **Copy Role Mappings**.
6. In Copy Role Mappings From Another Group, search for a group that you want to copy roles from.
7. Click the roles in the Copy Roles area to select or deselect them, and then click **Copy**.
8. Click **Save**.

Remove Application Roles from a Group

You can remove capabilities inherited by a group from the application roles mapped to it.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Security** under **Service Administration**.
3. On the Security page, under the Groups tab, select a group from the displayed list of groups or search for a group.
4. In the group details region, click the Application Roles tab.
5. Select one or more roles from the displayed list or search for application roles and select the applicable role.

6. Click **Remove Mapping**.
7. In Remove Role Mapping, click **Remove Mapping**.

Assign Users to a Group

When you assign users to a group, you create user memberships for the group. You can assign one or more users to one or more groups.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Security** under **Service Administration**.
3. On the Security page, click the Groups tab.
4. Search for a group or select from the list of groups displayed under the Groups tab.
5. Click **Assign Users**.
6. In Assign Users, search for a user or select from the list of users displayed in this dialog.
7. Select the check box for one or more users and click **Assign**.

Remove Users from a Group

You can remove one or more users from a group.

After you remove a user from a group, the applicable user may need to logout for the change to take affect. There could be a gap of around 30 seconds for the refresh to happen.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Security** under **Service Administration**.
3. On the Security page, click the Groups tab.
4. Search for a group or select from the list of groups displayed under the Groups tab.

   You see the users assigned to the group under the Users tab in the group details region.
5. Select the check box for one or more users and click **Remove User**.

Copy User Mappings to a Group

As a security administrator, you can copy the users mapped to an existing group to a custom group to quickly add users to the new custom group.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Security** under **Service Administration**.
3. On the Security page, click the Groups tab.
4. In the Groups list, search for a group or select the group to which you want to copy the users.
5. On the Groups tab, click **Copy User Mappings**.
6. In Copy User Mappings From Another Group, search for a group that you want to copy users from.

7. Under Copy Users, verify all the users who'll be copied to the custom group, and then click **Copy**.
8. Click **Save**.

Manage Application Roles

Application roles in Oracle NetSuite Analytics Warehouse consist of duty roles for objects security. You can map the ready-to use application roles to groups to define the permissions associated with the group. You can add or remove groups mapped to an application role.

Topics:

Create an Application Role

You can create custom duty and data roles to secure subject areas, front-end objects, and row-level data respectively when the predefined application roles don't meet your business needs.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Security** under **Service Administration**.
3. On the Security page, click the Application Roles tab and then click **New Application Role**.
4. In Create a New Application Role, enter the application role name and specify the role type as data or duty.
5. Click **Save**.

Assign Groups to Application Roles

Use these instructions to map application roles to groups.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Security** under **Service Administration**.
3. On the Security page, click the Application Roles tab.
4. Search for an application role or select from the displayed list of application roles.
5. In the application role details region, click **Assign Groups**.
6. In Add Group Mappings, search for a group and select it or select from the displayed list of groups.
7. Click **Save**.

Remove a Group Mapped to an Application Role

Use these instructions to remove a group mapped to an application role.

1. Sign in to your service.

2. In Oracle NetSuite Analytics Warehouse **Console**, click **Security** under **Service Administration**.
3. On the Security page, click the Application Roles tab.
4. Search for an application role or select from the displayed list of application roles.
5. In the application role details region, select the check box for a displayed group or search for a group and select it, and then click **Remove Group**.
6. In the Remove Group Mapping? dialog, click **Remove Group**.

Delete an Application Role

You can delete the custom application roles. Upon deletion of the custom application roles, Oracle NetSuite Analytics Warehouse deletes the mappings to the groups.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Security** under **Service Administration**.
3. On the Security page, click the Application Roles tab.
4. Search for an application role or select from the displayed list of application roles.
5. Click **Delete Application Role**.
6. In the Delete Application Role? dialog, click **Delete Role**.

Manage Data Access through Security Assignments

As a security administrator, you need to map data security assignments to users to enable data level access.

Use the Security Assignments tab on the Security page to search for the currently set up data security assignments. You may either search for all records or narrow your search to a specific security context, security value, or user. You can remove a security assignment that you had set up or add new security assignments to a user.

Topics:

Create a Security Assignment

Use these instructions to create a security assignment in a specific security context.

Security contexts are categories that contain values that you can secure a user for. For example, you can define which users have access to which "ledgers" or "departments". In this example, "ledgers" and "departments" are security contexts. Within "ledgers", you can have "ledger A", ledger B", or "ledger C" as values. You first select "ledger", then select a value such

as "ledger A", and then select the users to secure for "ledger A". The selected users can access "ledger A".

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Security** under **Service Administration**.

   You see the Security page.
3. On the Security page, click the Security Assignments tab.

   You see all users who have been granted the security assignments in a specific security context.
4. Click **New Assignment**.
5. In New Security Assignment, under **Select Security Assignments**, select a security context, and then search for a security value or select from the displayed list.Move the selected security assignments to the column on the right.
6. Under **Select Users**, search for a user and select the user and move the user to the column on the right.

   Users are filtered based on the role associated with that context.
7. Click **Add to Cart** and then click **View Cart**.
8. In Security Assignments, click **Apply Assignments**.

   You can grant this security assignment to other users as required. Bulk assignments may take some time to process. See the Security Activity tab for details.

Delete a Security Assignment

Use these instructions to delete a security assignment. When you delete a security assignment, Oracle NetSuite Analytics Warehouse removes all users associated with the security assignment.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Security** under **Service Administration**.

   You see the Security page.
3. On the Security page, click the Security Assignments tab.
4. Select a security assignment from the displayed list of assignments or search for a security assignment and select it.
5. Click **Delete Assignment**.

Remove Users from a Security Assignment

You can revoke the security assignment granted to one or more users.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Security** under **Service Administration**.

   You see the Security page.
3. On the Security page, click the Security Assignments tab.

4. Select a security assignment from the displayed list of assignments or search for a security assignment and select it.
5. In the security assignment details region, select the users from the displayed list of users or search for and select the users.
6. Click **Remove User**.
7. In Revoke User Assignment, click **Revoke Assignment**.

Manage Users for a Security Assignment

As a security administrator, you can manage users for existing data security assignments. In the Manage Users dialog, you can revoke users for an existing assignment or add new users for that assignment.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Security** under **Service Administration**.

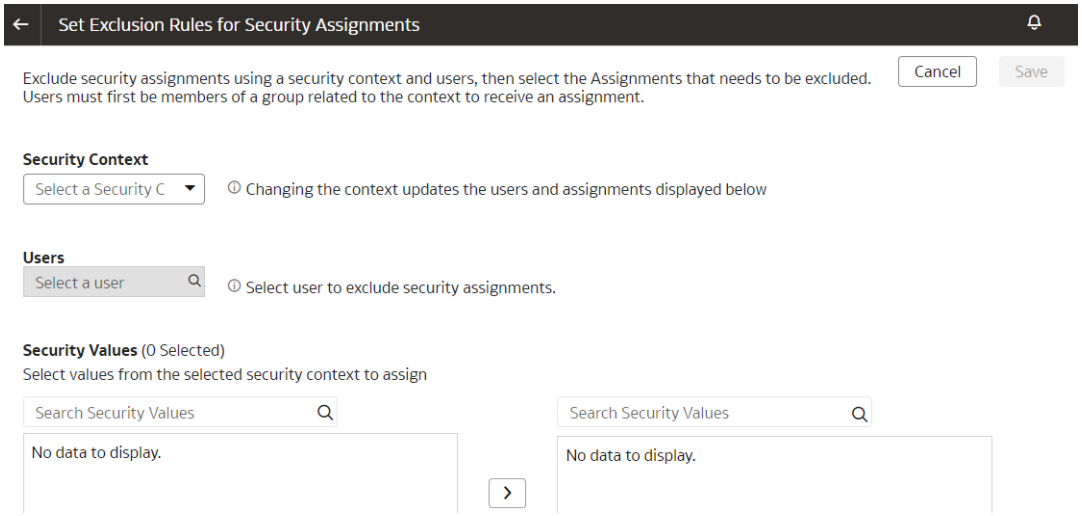
   You see the Security page.
3. On the Security page, click the Security Assignments tab.
4. Select a security assignment from the displayed list of assignments or search for a security assignment and select it.
5. In the security assignment details region, click **Manage Users**.
6. In Manage Users:
   a. Under **Add User**, search for a user and select the user.
   b. Under **User**, click the **Delete** icon to revoke the user from the assignment.
7. Click **Save**.

Set Exclusion Rules for Security Assignments

You can set up data security to exclude access for specific users within a security context for specific security assignments.

For example, you can grant access to all security assignments but the business unit ABC. This enables you to have a single rule for a single user within a security context. You can also remove the indirectly derived security assignments of the specific user. Ensure that the users for whom you want to exclude assignments are members of a group related to the security context. You can automate the application of the security exclusion rules by downloading the DataSecurityExclusionAssignments_csv.zip, making changes, and then uploading it; see Download and Upload Data Security Exclusion Rules.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Security** under Service Administration.
3. On the Security page, click **Security Assignments**, and then click **Exclusion Rules**.
4. On the Set Exclusion Rules for Security Assignments page, select the security context such as Ledgers in **Security Context**, select a user to exclude security assignments in **Users**, and then in **Security Values**, select the assignments that you want to exclude from the selected user within the selected security context.



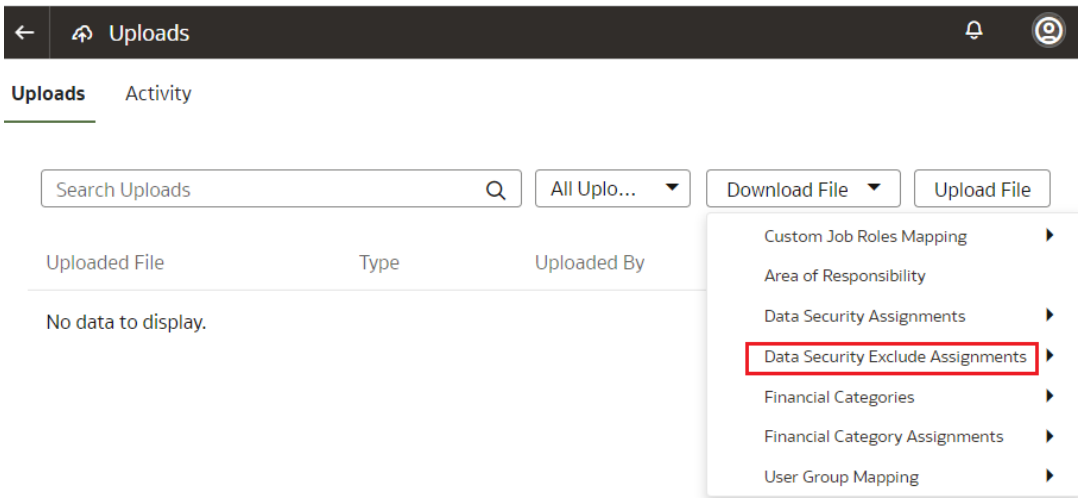
5. Click **Save**.

Download and Upload Data Security Exclusion Rules

If you want to automate the application of the security exclusion rules, then download the file to make the changes and upload it.

> **Note:**
>
> **Replace existing configuration settings** deletes existing data security assignments.

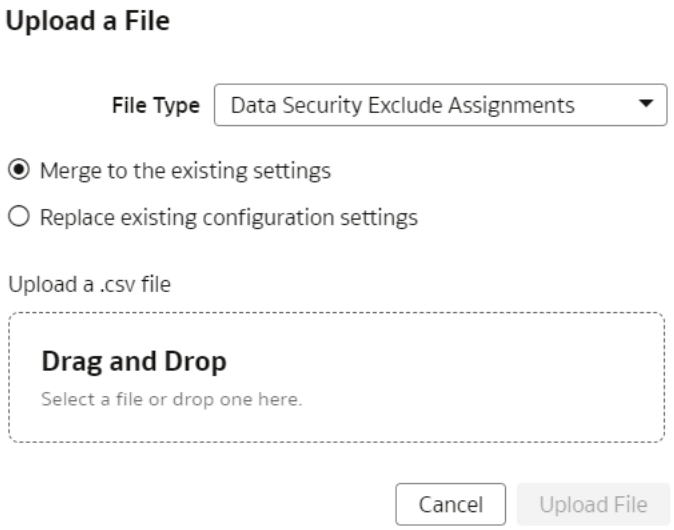
1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Uploads** under Application Administration.
3. On the Uploads page, click **Download File** and select **Data Security Exclude Assignments**, and then select the sample or current type of file that you want to download.



4. Unzip the DataSecurityExclusionAssignments_csv.zip file you downloaded and edit the csv file as required.

   When you're done updating the file, save your changes.

5. On the Uploads page, click **Upload File** and select **Data Security Exclude Assignments** in **File Type**.



6. Select whether you want to **Merge to the existing settings** or **Replace existing configuration settings**.
7. Select the file you want to upload and click **Upload File**.

   You can review the status of the upload on the Uploads tab.
8. Use **Actions** next to the file name to perform actions on a specific upload file:
   - Click **Properties** to check the upload statistics.
   - Click **Download** to download the file you just uploaded in the event you want to upload the file again.
   - Click **Delete** if a file fails to process and you want to remove the file history.

View Activity History

View all the security-related activities or filter them by object type and by date for security audit purposes.

The status icon next to each action shows whether it is in progress, in a warning state (if items in a bulk action failed), failed (and therefore incomplete), or completed successfully. You can also hover over error icons to read the full error message that might assist with troubleshooting.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Security** under **Service Administration**.

   You see the Security page.

3. On the Security page, click the **History** tab.

   You see all the activities by object type and date, by default. You can organize the display by sorting columns, searching for descriptive text, selecting a specific object type, or selecting the number of days from the drop-down lists.

4

Customize Oracle NetSuite Analytics Warehouse

You can customize the semantic model and groups.

As a modeler or modeler administrator, you can customize your semantic model. Customization enables you to make the data that you moved into the analytics warehouse more useful for reporting. As a security administrator, you can create, configure, and assign custom groups to users.

Topics:

About Semantic Model Customization

You can customize the semantic model to extend it for your business requirements. You extend the semantic model to make the data that you moved into the warehouse more useful for reporting.

Oracle NetSuite Analytics Warehouse currently provides these methods for extending the semantic model:

- Extend the Semantic Model Using the Sandbox Framework
- Extend the Semantic Model Using the Branch Framework

If your instance has the semantic model extensions Sandbox framework, then you can use the information about extending the semantic model using the Sandbox framework. If your instance has the semantic model extensions Branch framework, then you can migrate to the semantic model extensions Sandbox framework because the Branch framework is planned to be deprecated in a future release. See Migrate to the Sandbox Framework for Semantic Model Extensions.

Here's a comparison of the capabilities of the Branch and Sandbox framework for semantic model extensions:

Branch Framework	Sandbox Framework
Action centric view.	Model centric view.

Branch Framework	Sandbox Framework
Steps done one-by-one but are independent of each other.	Related steps are grouped together making it easier to complete a fully consistent unit of work.
Has the concept of branches that can contain steps that are unrelated.	Has the concept of a sandbox that can contain fully formed dimensional models or “stars”. All objects within a star are related.
No graphical view of the model.	Complete graphical view of each star that shows the ready-to-use and custom objects.
Publish branches to test. Merge to main when completed.	Publish sandbox to test, merge to main when completed.
Customization steps are disconnected. Different steps done on the same object can override each other in different branches.	All customizations done to an object are visible together. You see the result of all operations at any time.

The semantic model consists of these components:

- Oracle Content: This is the base model provided by Oracle. Your customizations are layered on this.
- System Extensions: Your descriptive flexfield and data augmentation changes are available in this component. See View Activity History of Semantic Model Extensions to know about the scenarios in which Oracle NetSuite Analytics Warehouse applies the system extensions.
- User Extensions: Your extensions are available in this component.
- Security Configurations: You can secure the objects of all the other components against the application roles in this component.

Oracle NetSuite Analytics Warehouse periodically evaluates the customizations and sends notifications to the users with Modeler Administrator and Modeler privileges to correct errors and warnings in the extensions as soon as possible to prevent errors during patching. If these errors aren't fixed and a patch is initiated (or a mandatory patch is auto-applied), then patching may encounter issues. In that case, customizations that haven't been merged to the main branch are removed and the factory semantic model is upgraded. You must reapply the applicable customizations after the patching is complete.

Recommendations and Tips to Extend the Semantic Model

Before extending your semantic model, review the recommendations and tips to ensure that your extensions work as expected.

Database Naming Standards for Autonomous Data Warehouse Objects

- Prefix a custom object with `X_ZZZ_` where `ZZZ` is an abbreviation of your organization.
- Suffix different objects as:
  - _A = Aggregate
  - _D = Dimension
  - _DH = Dimension Hierarchy
  - _F = Fact
  - _H = Helper
  - _M = Map Dimension

  - _MD = Mini Dimension
  - _V = Views
  - _MV = Materialized View
  - _DS = Data Augmentation Dataset
  - _EXT = Data Augmentation Extension
- Don’t create any table starting with "DW" in custom schemas and the OAX_USER schema because this may result in conflict with the prebuilt object names. If you create tables starting with "DW", then these tables won't show as custom tables in the Semantic Model Extensions wizard.

Deployment

- Migrations must flow in a single direction only. Choose one environment to be the master Development environment. After user acceptance testing, generate and deploy a Semantic Extensions bundle to migrate changes to Production and other environments.
- Don’t export the Semantic Extensions bundle separately and then do security promotion through test to production. Generate the Semantic Extensions bundle and include the extensions you want and then include security as well.

Data Augmentation Datasets and Flexfields

- Ensure that the changes in source are addressed in your Oracle NetSuite Analytics Warehouse instance. For example, if a descriptive flexfield used in a custom subject area has been disabled in the source, then you must replace or remove the applicable descriptive flexfield in Oracle NetSuite Analytics Warehouse else the applicable semantic model extension fails.
- You can reference synonyms from the data augmentation datasets in the semantic model extensions after the initial full load for the data augmentation has completed. Use the "Run Immediately" option in the data augmentation to execute the full load straightaway.

Extending

- When joining facts to dimensions, ensure that the columns being joined are of compatible data types.
- When extending DEGEN Dimensions ("Details" folders), always maintain the same level of granularity by joining on the Primary key(s) of the fact with a one to one [1:1] relationship. Don't define many to many [M:M] joins because it may cause performance degradation and data duplication.
- When creating a custom dimension, you may unselect "Add hierarchy to Subject Area". However, it is still necessary to define a Hierarchy Primary Key and Display Attribute. Click on Selected Data Elements Detail folder, then the Properties edit icon to define the Hierarchy Primary Key and Display Attribute.
- When extending a dimension (if the extension granularity is one to one [1:1] with the prebuilt dimension) combine multiple extensions for the same dimension in a single source (table/view/synonym) in Autonomous Data Warehouse. It is preferable to have one extension with many columns, rather than have multiple extensions per column.
- If it's necessary to have multiple extensions on the same dimension due to varying one to many (1:M) relationships (such as multi-select), then be cautious of an index length limitation that may be hit. To avoid the constraint, name the table/view/synonym as short as feasible. For example, FDI_X_SZ_V (Size) and FDI_X_PR_V (Price).

- When adding a custom fact, always set the content levels for the custom dimensions that are joining to the custom fact.
- When adding a custom hierarchy, avoid aiming to display the grand total levels in visualizations because custom hierarchies are exposed only from the first level. The prebuilt hierarchies too don't expose the total levels. The Grand Total level just gives the grand total amount; hence use it only when there is no join between a fact and dimension and the metric has to be set at a total level.
- When naming objects (dimensions, facts, and columns) remove all leading and trailing spaces. You can use underscores and spaces in names but avoid all other special characters.

Impact of Data Flows and Data Replication

If you're using data flows and data replication capabilities in Oracle Analytics Cloud associated with your Oracle NetSuite Analytics Warehouse instance and creating semantic model extensions in Oracle NetSuite Analytics Warehouse, note that it may impact the performance of Oracle Analytics Cloud. You can schedule data flows outside of business hours because running complex data flows during business hours may have a negative impact on the overall system performance.

Extend the Semantic Model Using the Sandbox Framework

The Sandbox framework provides better usability and understanding of the semantic model and customizations along with better performance and consistency of the model. The Sandbox capabilities result in a much faster time to have all changes available for reporting.

The Sandbox framework for the semantic model extensions adopts an object focused experience instead of a task focused experience in the existing capability. This enables you to see changes done to a specific objects rather than having to derive the changes from several scattered steps. The Sandbox capability has a logically organized flow for better usability with recommended practices enforced and better performance.

The Sandbox capability allows you to:

- Make changes within sandboxes, which are each user's work area.
- Publish a sandbox to test and merge it to the Main once tested.
- Keep all changes for an object available in one place.
- Publish only one sandbox at a time. Publishing a sandbox removes any sandbox already published. If you've merged a sandbox, then the system preserves the changes if you publish any other sandbox.
- Zoom and focus on specific areas of the logical star using the Graphic tab on the Logical Star page.
- Rearrange the objects in the logical star using the Graphic tab on the Logical Star page.
- View all the joins in a tabular format using the Tabular tab on the Logical Star page.

A typical workflow to create extensions involves these:

1. Create a sandbox.
2. Select **Perform Action** and then select **Create or Manage a Star** .
3. Make changes as required (the changes are done to the logical model).
4. Select **Perform Action** and then select **Manage Subject Areas**.
5. Incorporate logical changes in the desired subject areas.

6. Go back to Semantic Model Extensions page, select **User Extensions**, select **Publish Model**, and then select the sandbox to publish.
7. In Oracle Analytics Cloud associated with your Oracle Fusion Data Intelligence instance, verify if the changes are reflected in the subject area.

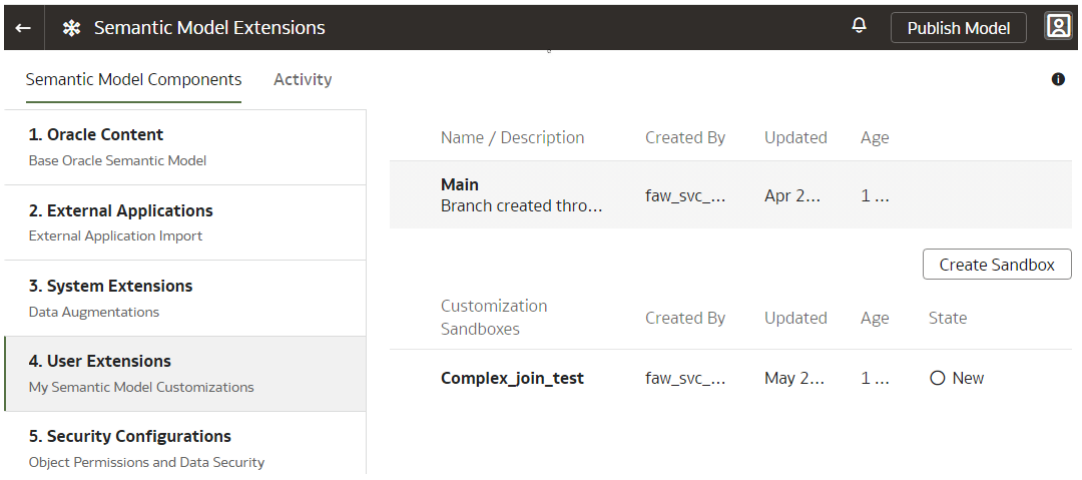
Topics:

Create Sandbox

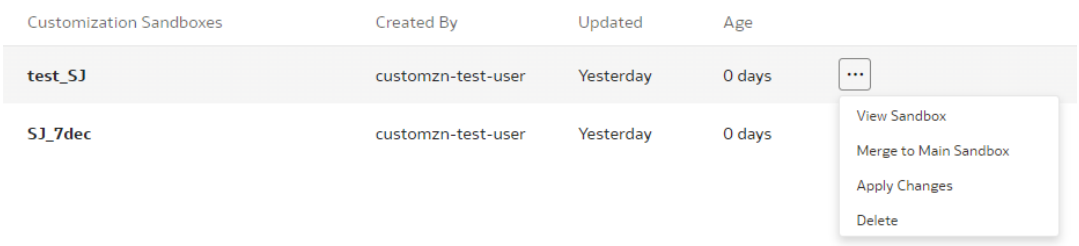
To begin customizing your semantic model, create a sandbox.

You add customizations to the production environment. After you have added and tested your customizations, you can publish them to the model in the production environment.

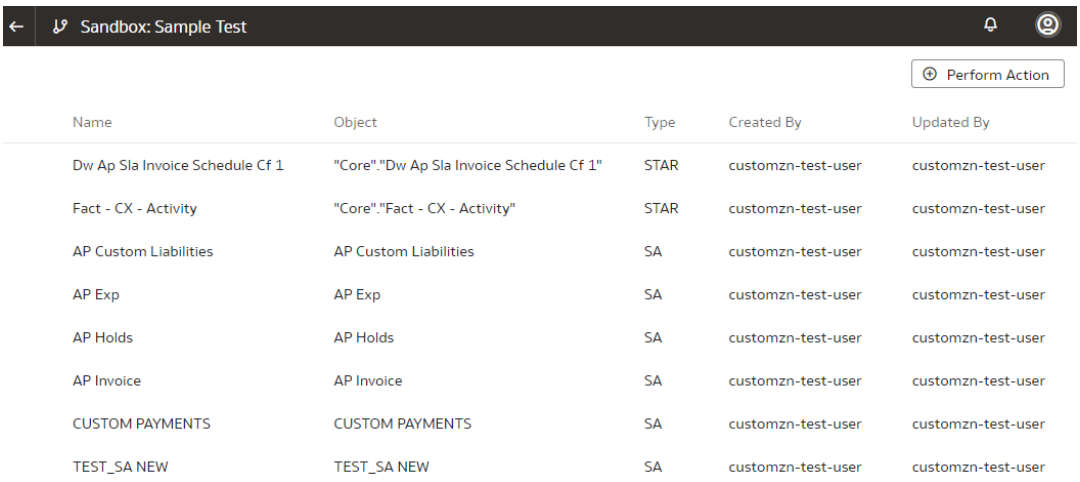
1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.
3. On the Semantic Model Extensions page, click **User Extensions**.
4. In the User Extensions region, for Customization Sandboxes, click **Create Sandbox** to create your customizations.



5. In Create a Sandbox dialog, enter a **Name** having up to 80 characters or less, provide a **Description**, and click **Done**.
6. On the Semantic Model Extensions page, under Customizations Sandbox, hover over an applicable sandbox to view **Actions**, and then click **View Sandbox**.



7. On the selected sandbox Details page, click **Perform Action**, and select as applicable.



See:

- Manage Subject Areas
- Manage Logical Star
- Manage Variables

Manage Subject Areas

The Manage Subject Areas action enables you to organize all entities and attributes available for reporting in subject areas.

You can create business-friendly names and organize them in a desired order within folders to make it easier to find and include in the reports. The typical organization is to have each dimension organized in a folder with all its attributes within it, followed by folder for facts and calculations. You can rearrange columns based on your organizational preferences.

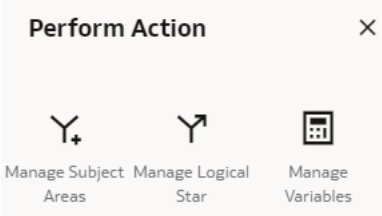
You can create a subject area or modify a subject area.

Create Subject Area

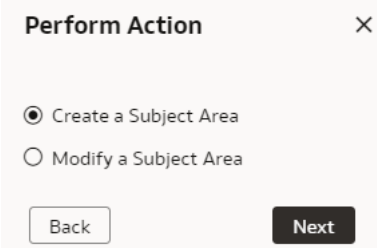
You can create a subject area as a container and later add facts and dimensions to your new subject area or create a subject area based on an existing one. The subject area enables you to organize all entities and attributes available for reporting.

1. On the Semantic Model Extensions page, under Customizations Sandbox, hover over an applicable sandbox to view **Actions**, and then click **View Sandbox**.

2. On the selected sandbox Details page, click **Perform Action**, and then select **Manage Subject Areas**.



3. In Perform Action, select **Create a Subject Area**, and then click **Next**.



4. In step 1 of the wizard, create a subject area using one of the methods:
   - Select **Create a Subject Area** to create a subject area container, and provide these details:
     a. Enter a name without any leading or trailing white spaces, add a description, and then click **Next**.
     b. In step 2 of the wizard, click **Manage Elements**, and then click either **Manage New Customizations** to select custom elements that you created or **Manage Factory Customizations** to select factory data elements to rearrange the subject area elements that are delivered by Oracle.
     c. Click **Add Subject Area** to select and add data elements from multiple subject areas.
     d. In step 3 of the wizard, organize and rename the data elements in your new subject area, and then click **Next**.
     e. In step 4 of the wizard, review your new subject area and click **Finish** to create it.
   - Select **Create a Subject Area based on an existing one** to create a subject area using an existing one in the system, select an existing subject area, name your subject area, and then click **Next**. Complete steps 2, 3, and 4 of the wizard.

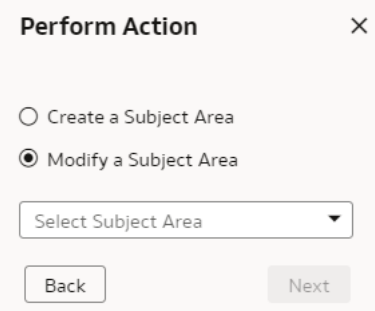
Modify Subject Area

You can modify custom and prebuilt subject areas. Modify a custom subject area to change the previously selected data elements or add more data elements and modify a prebuilt subject area to add more data elements.

Ensure to create custom elements if you want to add them to the custom subject area and add additional columns to the prebuilt subject area prior to modifying either of them.

1. Sign in to your service.

2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.
3. On the Semantic Model Extensions page, click **User Extensions**.
4. On the Semantic Model Extensions page, under Customizations Sandbox, hover over an applicable sandbox to view **Actions**, and then click **View Sandbox**.
5. On the selected sandbox Details page, click **Perform Action**, and then select **Manage Subject Areas**.
6. In Perform Action, select **Modify a Subject Area**, select a custon or prebuilt subject area that you want to modify, and then click **Next**.



7. In step 1 of the wizard, view the selected subject area details and click Next.
8. In step 2 of the wizard,
   - If you're modifying a custom subject area, click **Manage Elements** and then click **Manage New Customizations** to select custom elements that you created.
   - If you're modifying a prebuilt subject area, click **Manage Elements** and then click **Manage Factory Customizations** to select the prebuilt elements that you extended.
9. Click **Finish**.

Manage Logical Star

A logical star is the basic complete unit of a dimensional model with a fact at the center and joined to the surrounding dimensions. Manage a logical star by adding and updating objects, attributes, joins, and calculations.

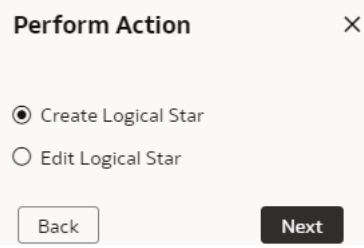
Facts contain elements that you can measure such as count, aggregate, and perform statistical operations on; while dimensions contain elements that provide context to those measurements. Each logical star has one fact and one or more dimensions. You can manage your own custom star or you can manage a prebuilt star by adding dimensions. You do these operations to extend the model to make use of custom data objects or elements that you've added to the warehouse or to create new calculations or joins to address your reporting needs.

Create Logical Star

Create a custom logical star to use custom data objects or elements that you have added to the warehouse or to create new calculations or joins to address your reporting needs.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.

3. On the Semantic Model Extensions page, click **User Extensions**.
4. On the Semantic Model Extensions page, under Customizations Sandbox, hover over an applicable sandbox to view **Actions**, and then click **View Sandbox**.
5. On the selected sandbox Details page, click **Perform Action**, and then select **Manage Logical Star**.
6. In Perform Action, select **Create Logical Star**, and then click **Next**.



You're now ready to add facts, dimensions, hierarchy, and additional columns.

Add Fact

Add elements that you can measure such as count and aggregate, and perform statistical operations to your custom logical star using the Add Fact option.

While selecting an aggregation rule for each fact column to set the aggregation behaviour, use a time-balanced aggregation when the added measure mustn't be "aggregated" by default across a time dimension. Oracle Fusion Data Intelligence supports non-aggregation types like "Last" or "First" in place of the "SUM" aggregation type when required. Use a level-based aggregation when the underlying measure must always be calculated to a specific level of a predefined dimensional hierarchy. For example, in a product hierarchy that has the Product Total, Product Category, Product Sub-Category, and Product Details levels, you add a new measure called "Revenue" and need this "Product Category Revenue" measure to be aggregated to Product Category, then you must use the level-based aggregation and choose the right level of the Product Dimension. This setting enables Oracle Fusion Data Intelligence to always aggregate and show the value of the measure at the Product Category level. This is useful when you need to calculate Product Revenue as a % of Category Revenue.

1. Navigate to the Semantic Model Extensions page, under Customizations Sandbox, hover over an applicable sandbox to view **Actions**, and then click **View Sandbox**.
2. On the selected sandbox Details page, click **Perform Action**, select **Manage Logical Star**, and then select **Create Logical Star**.
3. On the Logical Star page, click **Add Fact**, and in step 1 of the Add a Fact wizard, select the schema, and then select a view or table or synonym as the object. For example, FCT_CALC_Extensions.

   You see the fact table for the selected object.
4. In the details of the fact table for the selected source table, click the **Select Fact** and **Use for Key** check boxes for the source columns that you want to add to your new fact table in the target subject area.
5. Optional: In the details of the fact table for the selected source table, under **Select Degen Attribute**, click the check boxes for the attributes for which you need the degenerate dimension to be created.
6. If any of the selected attributes have been removed or modified in the source table since the last refresh, then you see such columns highlighted and a message asking whether

you want to update the table. Select **OK** in the message to reload the source columns. If you want to review the changes to the source columns, then click **Cancel** in the message, and later click **Refresh** to reload the source columns.

If any of the attributes that you haven't selected have been removed or modified in the source table, then you see the refreshed list of source columns. If any of the custom columns fail validation during the refresh, then you see a message asking you to resolve the cause of failure and revalidate.

7. Optional: Click Create Column to add a new column to your new fact table in the target subject area using these instructions:
   a. In Create Column, enter a display name.
   b. Under **Data Elements**, search for a data element from the physical table of the selected dimension table.
   c. From the search results, double-click the data element to place it in the text pane.
   d. Under **Functions**, search for a function to construct a column using expressions. For example, search for functions like "substring" or "concatenate" to construct new expression-based columns. From the search results, double-click the applicable result to add it to the central text pane.
   e. Click **Validate**, and then click **Save**.
8. Select the aggregation rule for each fact column to set the aggregation behaviour. You can set the time-balanced aggregation rule for a time dimension and hierarchy level-based aggregation rule for a dimension using these steps:
   a. For a fact column, click the **Time-Balanced Aggregation** icon.
   b. In the Time-Balanced Aggregation dialog, click **Add Time Dimension**, adjust the aggregation rule, and then click **OK**.
   c. For a fact column, click the **Hierarchy Level-Based Aggregation** icon, select the dimension and level. Click **Add Dimension** to add more dimensions. Click **OK**.

Manage Dimensions

You can create a custom dimension, join it to the prebuilt or custom facts, and add the custom dimension to any subject area to meet your business requirements.

You can add a new or an existing dimension.

1. Navigate to the Semantic Model Extensions page, under Customizations Sandbox, hover over an applicable sandbox to view **Actions**, and then click **View Sandbox**.
2. On the selected sandbox Details page, click **Perform Action**, select **Manage Logical Star**, select **Edit Logical Star**, select the applicable fact and then select **Next**.
3. On the Logical Star – Fact page, click **Manage Dimension**, and select either **Add Dimension** or **Add Existing Dimension**.

Add Dimension

Add dimensions to facts to complete a new star or to update existing stars. You can create joins from your custom dimension to a prebuilt fact.

1. In step 1 of the Add a Dimension wizard, select the schema, and then select the dimension table in **Object**. For example, COST_CENTER_VIEW1, and add a name in **Dimension Name**.

> **Note:**
>
> If you don't see the schema or table, then ensure that you have granted select permission to the OAX$OAC schema in the autonomous data warehouse. For example, `grant select on <schema>.<table>` to OAX$OAC. See Load Customization Data to the Autonomous Data Warehouse.

You see the attributes available in the selected dimension table. You can use the **Search** and **Filter** fields to limit the attributes displayed for the dimension table.

2. Select the attributes that you want to use from the dimension table and indicate an attribute to be used as the key for joining with a fact table in the target subject area.
3. If any of the selected attributes have been removed or modified in the source table since the last refresh, then you see such columns highlighted and a message asking whether you want to update the table. Select **OK** in the message to reload the source columns. If you want to review the changes to the source columns, then click **Cancel** in the message, and later click **Refresh** to reload the source columns.

   If any of the attributes that you haven't selected have been removed or modified in the source table, then you see the refreshed list of source columns. If any of the custom columns fail validation during the refresh, then you see a message asking you to resolve the cause of failure and revalidate.
4. Optional: Click **Create Column** to add another column to your dimension table in the target subject area using these instructions:
   a. In Create Column, enter a display name.
   b. Under **Data Elements**, search for a data element from the physical table of the selected dimension table.
   c. From the search results, double-click the data element to place it in the text pane.
   d. Under **Functions**, search for a function to construct a column using expressions. For example, search for functions like "substring" or "concatenate" to construct new expression-based columns. From the search results, double-click the applicable result to add it to the central text pane.
   e. Click **Validate**, and then click **Save**.
5. In step 2 of the wizard, assemble the product hierarchy using the attributes from this dimension and click **Finish**. See Add Hierarchy.
6. On the Logical Star: Fact page, in the Graphic tab, click on the prebuilt fact and drag drop on the custom dimension that you created to open the Join dialog.
7. In the Join dialog, select the join type, and then select the dimension keys to join them with the extended dimension keys. If you want to provide expressions as join conditions, then click **Complex Join** and in Create Joins, select applicable **Content Level**, click **Add Joins**, select the target and source logical tables, enter the join condition as an expression, and then click OK.

Add Existing Dimension

If you want to provide additional context to facts, you can create your own dimension and join to an existing available column in a fact.

For example, if you want to report on invoice categories, create a dimension called "Invoice Category" and join to a column in the fact that has that information. It is important to remember

that one dimension record must join to one or more fact records; it should be a 1-many join. You shouldn't have many to one or many to many joins between a dimension and fact table.

1. On the Logical Star – Fact page, click **Manage Dimension**, and select **Add Existing Dimension**.
2. In Add Table, select the dimensions to add.

Manage Extensions

After adding the extension, you can extend the dimensions, add hierarchy, and add columns to ensure that your custom logical star meets your business requirements..

On the Logical Star – Fact page, right click on an extension, click **Manage Extension**, and select any of these:

- Extend Dim
- Add Hierarchy
- Add Column

Extend Dimension

Extend prebuilt dimensions with additional attributes from another data source. For example, you can create a category column that isn't available in the prebuilt dimensions.

1. In step 1 of the Extend a Dimension wizard, select a schema and table from the database.
2. Select the columns that you want to expose or use as a key for creating the join.
3. Click in the **Display Name** table field to enter a new name for the column or to edit an existing one and then click **Enter** to accept or click **Esc** to cancel.
4. If any of the selected attributes have been removed or modified in the source table since the last refresh, then you see such columns highlighted and a message asking whether you want to update the table. Select **OK** in the message to reload the source columns. If you want to review the changes to the source columns, then click **Cancel** in the message, and later click **Refresh** to reload the source columns.

   If any of the attributes that you haven’t selected have been removed or modified in the source table, then you see the refreshed list of source columns. If any of the custom columns fail validation during the refresh, then you see a message asking you to resolve the cause of failure and revalidate.
5. Optional: Click **Create Column** to add another column to your dimension table in the target subject area using these instructions:
   a. In Create Column, enter a display name.
   b. Under **Data Elements**, search for a data element from the physical table of the selected dimension table.
   c. From the search results, double-click the data element to place it in the text pane.
   d. Under **Functions**, search for a function to construct a column using expressions. For example, search for functions like "substring" or "concatenate" to construct new expression-based columns. From the search results, double-click the applicable result to add it to the central text pane.
   e. Click **Validate**, and then click **Save**.
6. Click **Next**.
7. In step 2 of the wizard, select the join type, and then select the dimension keys to join them with the extended dimension keys. If you want to provide expressions as join conditions,

then click **Complex Join** and in Create Joins, click **Add Joins**, select the target and source logical tables, enter the join condition as an expression, and click **OK**.

8. Click **Finish**.

Add Hierarchy

Assemble the product hierarchy using the attributes from a dimension table. Hierarchies enable you to define aggregations and drill downs. This makes it easier to report on summary level and drill into details easily and within the same visualization.

1. In step 1 of the Add a Hierarchy wizard, name your hierarchy in **Hierarchy Name**.
2. Select, drag, and drop available data elements into the Selected Data elements pane to design a hierarchy for the dimension.
3. In the Selected Data Elements pane, click a level to update its primary key and set its display attribute in the Properties pane.

   You can add multiple levels in your hierarchy by right-clicking at a level and selecting Add Child or Add ‘n’ Child Levels. For example, your Region Hierarchy can have Region Total at Level 1, Region at Level 2, Country at Level 3, State at Level 4, and City at Level 5.
4. Ensure **Add hierarchy to Subject Area** is selected and click **Finish**.

Add Columns

You can create columns to provide additional data elements or calculations.You can add derived and physical columns.

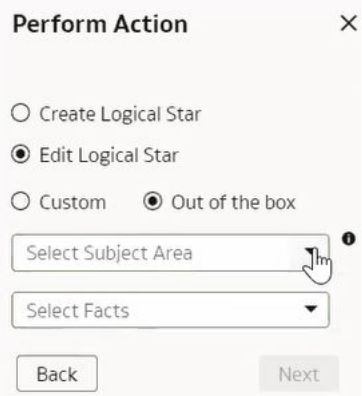
1. On the Add Column page, select **Add Derived Column**, and complete these steps:
   a. In Create Column, enter a display name.
   b. Under **Data Elements**, search for a data element from the physical table of the selected dimension table.
   c. From the search results, double-click the data element to place it in the text pane.
   d. Under **Functions**, search for a function to construct a column using expressions. For example, search for functions like "substring" or "concatenate" to construct new expression-based columns. From the search results, double-click the applicable result to add it to the central text pane.
   e. Click **Validate**, and then click **Save**.
2. On the Add Column page, select **Add Physical Column**, and complete these steps:
   a. In Select Physical Column, select the columns and click **OK**.
   b. On the Add Columns page, for the physical columns, select the **Display** check box to expose the columns, and click the **Logical Level** icon to set the required level.
   c. In Set Logical Level, select the dimension, select the level of the dimension hierarchy, and then click **OK**.

Edit Logical Star

Edit your logical star to modify any of the extensions that you had previously added or to add further extensions.

1. Navigate the Semantic Model Extensions page, under Customizations Sandbox, hover over an applicable sandbox to view **Actions**, and then click **View Sandbox**.

2. On the selected sandbox Details page, click **Perform Action**, and then select **Manage Logical Star**.
3. In Perform Action, select **Edit Logical Star**'
4. To select the prebuilt objects, select **Out of the box**, select a subject area and an applicable fact within the selected subject area, and then click **Next**. Select **Custom** to add custom objects to the logical star, select the applicable fact, and then click **Next**.



5. On the Logical Star – Fact page, click **Manage Dimension** and proceed with the steps discussed in Manage Dimensions.

Manage Variables

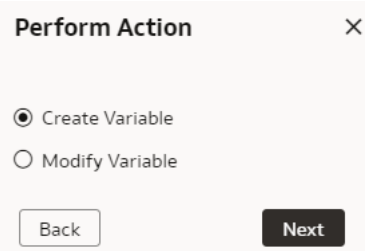
Use the Manage Variables action to control the behaviour of sessions and queries. You can create and modify the custom variables.

Create Variable

Create custom session variables that you can use in your semantic model.

The SQL query that you define is executed by user OAX$OAC. If you're using another schema in the query, then you must mention the schema name as prefix. You must ensure to grant user OAX$OAC access to all the database objects used in the query.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.
3. On the Semantic Model Extensions page, click **User Extensions**.
4. On the Semantic Model Extensions page, under Customizations Sandbox, hover over an applicable sandbox to view **Actions**, and then click **View Sandbox**.
5. On the selected sandbox Details page, click **Perform Action**, select **Manage Variables**, and then select **Create Variable**.



You see the wizard sequence to add the session variables and a list of existing session variables.

6. In step 1 of the wizard, check if any of the existing session variables serve your purpose. If yes, then you can exit the wizard and use the applicable existing session variables in your analyses. If no, then continue with the next steps to create the session variables that you require.
7. In **Initialization Block Name**, enter a name such as `Add a Session variable using Invoice Received Date`, add a brief description, and select a preceding initialization block in **Preceding Block**.
8. In **SQL Query**, enter the SQL query that would be executed in the autonomous data warehouse and return a value that you can use in the reports and click **Next**. For example, if you want to get the Exchange Rate Type that's defined in the system into a session variable, then you can use the following SQL script:

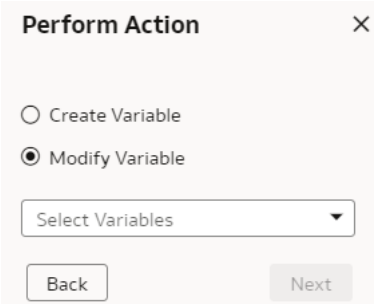
```
Copy
        SELECT PARAMETER_VALUE FROM DW_CONTENT_PARAM_CONFIG WHERE
        PARAMETER_CODE='PARAM_GLOBAL_EXCHANGE_RATE_TYPE'
```

9. In step 2 of the wizard, create the session variables using the output of the initialization block created in step 1 of the wizard. Select **Row-wise Initialization** to reset variable value for each row and **Use caching check boxes** to improve performance.
10. Click **Finish**.

Modify Variable

Modify a custom variable to update the SQL query that would be executed in the autonomous data warehouse and return a value that you can use in the reports.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.
3. On the Semantic Model Extensions page, click **User Extensions**.
4. On the Semantic Model Extensions page, under Customizations Sandbox, hover over an applicable sandbox to view **Actions**, and then click **View Sandbox**.
5. On the selected sandbox Details page, click **Perform Action**, and then select **Manage Variables**.
6. In Perform Action, select **Modify Variable**, select the variables that you want to modify, and then click **Next**.

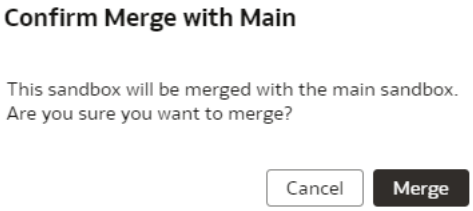


7. Follow through the wizard to modify the variable and click **Finish**.

Merge Customization Sandbox to Main Sandbox

After creating the semantic model extensions, you must merge the customization sandbox that contains your semantic model extensions into the main sandbox to make the extensions available for processing.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.
3. On the Semantic Model Extensions page, click **User Extensions**.
4. On the Semantic Model Extensions page, under Customizations Sandbox, hover over an applicable sandbox to view **Actions**, and then click **Merge to Main Sandbox**.
5. In Confirm Merge with Main, review the message and click **Merge**.



Apply Changes

After merging your semantic model extensions with the main sandbox, you must apply the changes to your semantic model to use the extensions in your visualizations.

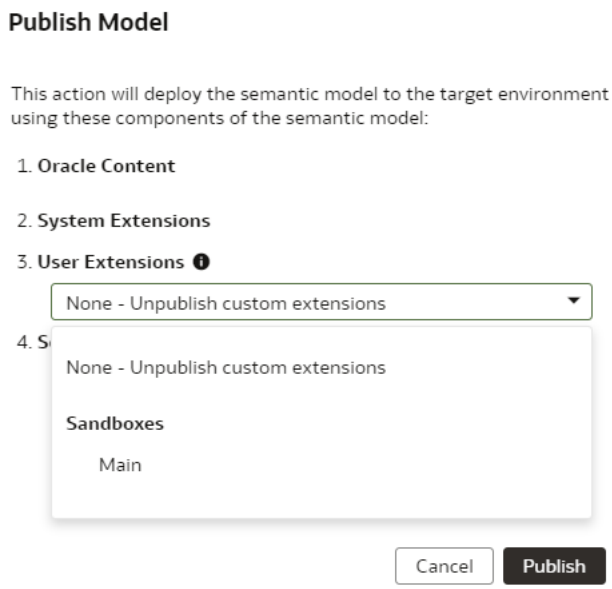
1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.
3. On the Semantic Model Extensions page, click **User Extensions**.
4. On the Semantic Model Extensions page, under Customizations Sandbox, hover over an applicable sandbox to view **Actions**, and then click **Apply Changes**.

Publish Model

You can publish the sandbox in the non-production environments such as development or test to ensure that there are no errors.

While publishing the data model, you can select the user extensions and security configurations that you added as part of customizing the semantic model. If you select the security configurations, then the system applies them on the user extensions that you selected. If the security configurations refer to elements in the model that aren't part of the user extensions, then the system excludes them at the time of publishing the model.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.
3. On the Semantic Model Extensions page, click **Publish Model**.
4. In Publish Model, select the user extensions and security configurations that you want to publish.



5. Click **Publish**.

Extend the Semantic Model Using the Branch Framework

You customize your semantic model in the non-production environments such as development or test by creating branches and adding steps to those branches.

> **Note:**
>
> Oracle recommends that you migrate to the semantic model extensions Sandbox framework because the Branch framework is planned to be deprecated in a future release.

You use a branch or version to publish your changes to the model. You can apply the model from the branch to a local service instance for testing. When the changes are correct, you can merge that branch with the main branch. You can merge multiple branches with the main branch over time. When you have a set of changes finalized, you can version the main branch and use the Bundles functionality to move the customizations to another environment. See the information regarding the Semantic Extensions Bundle in Bundle Your Application Artifacts.

You can copy the steps from the main branch, edit them directly, and later merge them with the main branch. However, you must ensure not to perform two or multiple levels of copies. For example, you must not copy steps from the main branch to another branch, then copy the steps over to yet another branch, and then merge the second branch with the main branch.

You can customize the model by extending prebuilt dimensions with additional attributes from another data source, by adding a fact to an existing subject area, and by reorganizing the prebuilt subject areas to create a new subject area to name a few. If you've merged an external application, then note that both external semantic model and semantic model extensions can co exist. On the Publish Model page, select "Yes" if you want the external semantic model to be included.

Don’t maintain changes in a local branch for long periods of time. Local branches can become out of synchrony when changes to the Oracle NetSuite Analytics Warehouse environment occur. Use a local branch for development, then merge to the main branch after user acceptance testing. Promote the main branch from Development to Production environments using the Semantic Extensions bundle. Whenever a system extension run is replaying the master branch, you must not edit the main branch during this time as that might cause a conflict leading to failure of the semantic extensions.

Topics:

Migrate to the Sandbox Framework for Semantic Model Extensions

Prior to switching over to the Sandbox framework for semantic model extensions, note the tasks and ensure to take appropriate action where required in the suggested sequence.

Begin by migrating the non-production instances to the Sandbox framework before migrating the production instance.

Topics:

Perform the Pre-Migration Tasks

Prior to the migration, note these and take appropriate action:

1. Ensure that you schedule or apply all the release updates prior to the migration to avoid not being able to rollback.

   The option to revert to the Branch framework gets disabled after 60 days from the migration date or an Application update, if it happens first post migration to the Sandbox framework.
2. As a pre-clean up task, create a Semantic Model Extensions bundle of the published branch prior to making any semantic model extension changes.
3. Delete or enable disabled steps in the Main branch to be migrated because the migration process migrates only the enabled steps.
4. Delete or merge all unpublished branches because these won't be migrated.
5. Merge the published branch with the main branch.
6. Publish the Main Branch, confirm that all steps are error free, and confirm that all the subject areas and security referenced in workbooks operate as expected.
7. As a post-clean up task, create a Semantic Model Extensions bundle of the published Main Branch.

Validate the Main Branch

On the Semantic Model Extensions page, when you see the banner to schedule the migration, don't click **Schedule now** until you've validated the main branch successfully with no errors.

Note these and take applicable action prior to initiating the validation process:

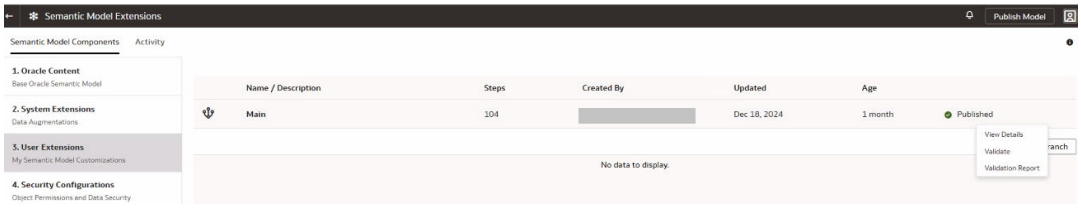
- Don't perform any development or administrative activities that effect the semantic model extensions during validation or while the migration is in process. This includes modifying existing data augmentation or creating new data augmentation that have the semantic model extensions.
- Remove leading and trailing spaces from the object names.
- Remove special characters such as parenthesis from the object names.
- Ensure all keys are present on the join columns.
- Confirm all referenced source database objects and columns exist in Autonomous Data Warehouse.

- Confirm grants are applied on all referenced source database objects using

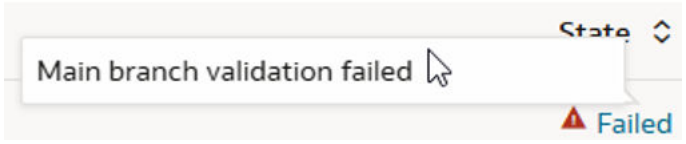
```
GRANT SELECT ON "OAX_USER"."TABLE_VIEW" TO "OAX$OAC";
```

- Resolve any duplicate object names.

1. On the Semantic Model Extensions page, click the Actions Menu for the Main Branch, and then click **Validate** to produce the Validation report.



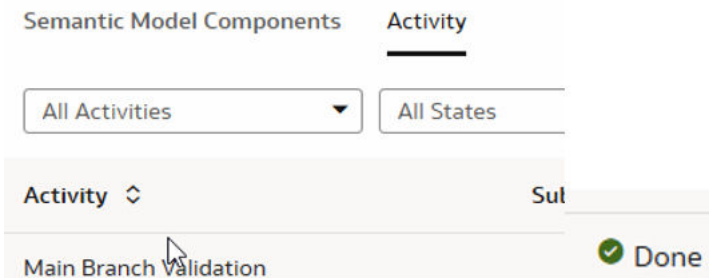
2. Click the Activity tab on the Semantic Model Extensions page and wait for the validation of the main branch to complete. If the **State** is **Failed**, then don't proceed with the migration until you've resolved all of the errors.



3. Click the Actions Menu for the Main Branch, and then click **Validation Report** to download and review the validation report.

> **Note:**
>
> The validation report is enabled only if the validation fails with errors.

4. Resolve all errors in the validation report and repeat Validate and Validation Report steps until you've resolved all errors. For any errors unresolvable, contact Oracle Support before proceeding.
5. Click the Activity tab on the Semantic Model Extensions page and wait for the validation of the main branch to complete. If the **State** is **Done** and the report is error free, then proceed with scheduling the migration.



> **Note:**
>
> Don't proceed with the migration until all errors are resolved.

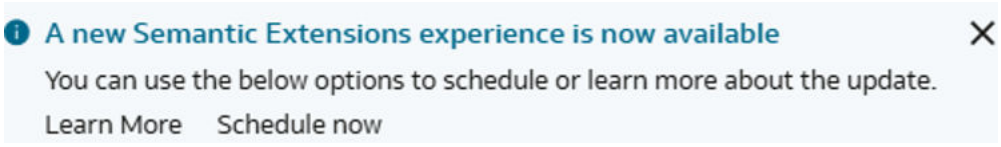
Schedule the Migration

You can initiate migration of your extensions to the Sandbox framework by submitting the request from the banner message on the Semantic Model Extensions page when you click **User Extensions**.

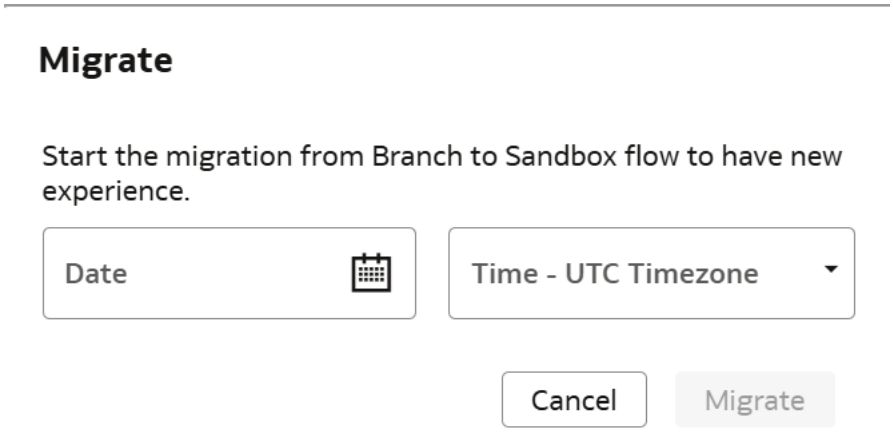
Set up the migration to happen automatically by scheduling it from the banner message. Ensure that you schedule the migration during a low usage period to avoid reporting outages.

> **Note:**
>
> You can't use the semantic model extensions capability during the migration process.

1. On the Semantic Model Extensions page, in the banner announcing the ability to schedule the migration, click **Schedule now**.



2. In the Migrate dialog, specify the date and time for the migration to begin.



3. Track the progress of the migration in the Activity tab on the Semantic Model Extensions page when the migration starts.

Perform the Post Migration Verifications

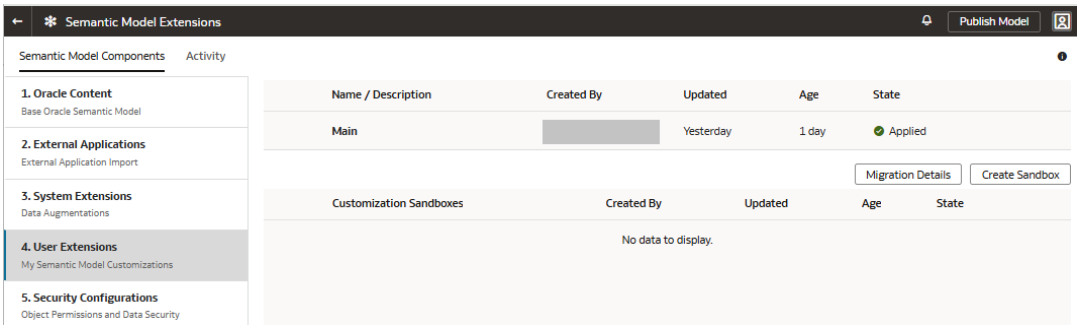
After the migration is complete, note these and take appropriate action:

The migration process converts the single Main Branch to a single Semantic Model Extensions Sandbox. It migrates the existing customizations to the new framework. The security configurations remain unchanged in the new framework, hence no action from you is required.

1. On the Semantic Model Extensions page, click the Activity tab and monitor the status of the migration.

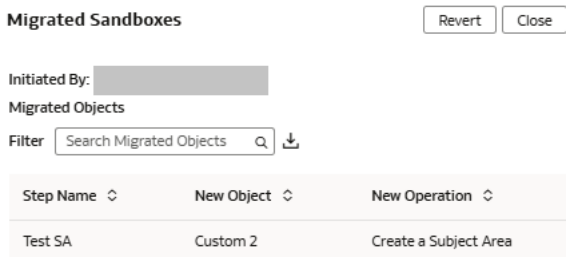
   Once the migration is complete, you receive a notification on the Console. If the migration fails, click the **Failed** state in the Activity tab for the applicable task and resolve the issues. For any unresolvable issues, reach out to Oracle Support.

2. After the migration is completed successfully, on the Semantic Model Extensions page, click **Migration Details**.



3. In Migrated Sandboxes, click the download icon to download the migration report.

   This report shows which customizations are available for which object with the Main branch object to the Main sandbox object mapping. This enables you to see all your customizations that have been brought over. In most cases, operations like Add Dim, Add Fact, Add Variable, Extend Dimension, and Add Columns have 1:1 correspondence between the old and the new framework. The Main sandbox is published and ready for use. All your subject areas, visualizations, and reports remain unchanged.



4. Sign in and sign out of your service after the migration completes. Confirm that all the subject areas and security referenced in the workbooks operate as expected.

Revert to the Branch Framework for Semantic Model Extensions

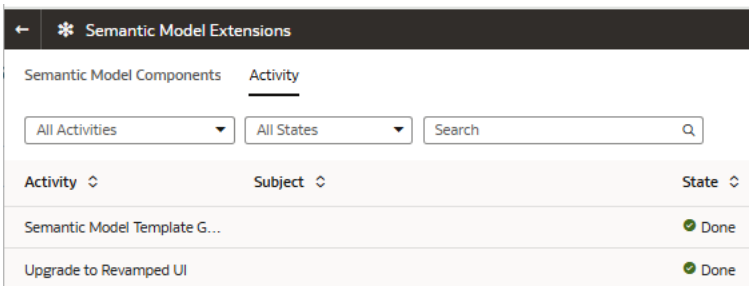
Post migration, an option to revert to the Branch framework if customizations aren't as expected is available. The migration rollback option gets disabled after 60 days from the

migration date or an Application update, if it happens first post migration to the Sandbox framework.

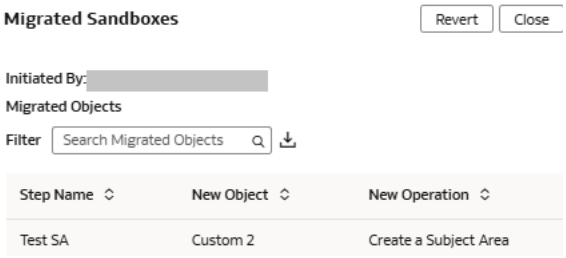
Note these:

- Ensure to complete the reversal process within this period if you must revert. If you choose to revert, then the semantic model rolls back to the state prior to the migration being initiated. You can't rollback the migration if you've completed an Application update after completion of the migration process.
- Oracle maintains a backup of the existing semantic model extensions at the time the migration is initiated. If you encounter issues during or after migration, Oracle uses the backup to troubleshoot and enable required extensions on the new framework. Should you choose to revert to the previous state, the semantic model is restored from this backup.
- If you've made further changes to the semantic model in the new framework before choosing to revert, these customizations can't be migrated to the previous semantic model extensions framework.

1. On the Semantic Model Extensions page, click the Activity tab and verify that the status of the scheduled migration task **Upgraded to Revamped UI** is **Done**.



2. On the Semantic Model Extensions page, click **Migration Details**.
3. In Migrated Sandboxes, click **Revert** to rollback the migration.



Create a Branch

To begin customizing your semantic model, create a branch of the main semantic model.

You add customizations to the production environment. After you have added and tested your customizations, you can publish them to the model in the production environment.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.

3. On the Semantic Model Extensions page, click **User Extensions**.

   You see the main and existing customization branches.

4. In the User Extensions region, for Customization Branches, click **Create Branch** to create a branch as an empty container.

5. In Create a Branch, enter a name for your branch, for example, `Add Cost Center`.

6. Optional: Add a description and click **Done**.

   You see the Add Step dialog. You can continue to add the steps or you can add steps to the new branch container later using the **Add Step** button. See Add a Step to a Branch. You see the new branch on the Semantic Model Extensions page under **Customization Branches**.

Edit a Branch

Before you apply a branch to the main branch of your semantic model, you can edit the branch description to make it more meaningful.

1. Sign in to your service.

2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.

3. On the Semantic Model Extensions page, click **User Extensions**.

4. In the User Extensions region, under Customization Branches, hover over a branch to view**Actions**.

   You see the actions that are applicable to the branch.

5. From **Actions**, click **Edit Description** to update the branch description, and then click **Done**.

Add a Step to a Branch

You can add customization types such as "Extend a Dimension" as a step to an existing unapplied branch or a new branch that you create.

1. Sign in to your service.

2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.

3. On the Semantic Model Extensions page, click **User Extensions**.

   You see the main and existing customization branches.

4. In the User Extensions region, under Customization Branches, click a branch to open the Branch page.

5. On the Branch page, click **Add Step**.

6. In Add Step, select a customization type such as **Extend a Dimension**.

   You see the wizard sequence to add details for the selected customization type.

Add a Dimension

You can create a custom dimension, join it to the prebuilt or custom facts, and add the custom dimension to any subject area to meet your business requirements.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.
3. On the Semantic Model Extensions page, click **User Extensions**.

   You see the main and existing customization branches.
4. In the User Extensions region, under Customization Branches, click a branch to open the Branch page.
5. On the Branch page, click **Add Step**.
6. In Add Step, select **Add a Dimension**.

   You see the wizard sequence to add a dimension.
7. In step 1 of the wizard, enter a name for your customization step, for example, `Add Point of Sale Dimension` and add a brief description.
8. In step 2 of the wizard, select the schema, and then select the dimension table in **Object**. For example, `COST_CENTER_VIEW1`.

   > **Note:**
   >
   > If you don't see the schema or table, then ensure that you have granted select permission to the OAX$OAC schema in the autonomous data warehouse. For example, `grant select on <schema>.<table> to OAX$OAC`. See Load Customization Data to the Autonomous Data Warehouse.

   You see the attributes available in the selected dimension table. You can use the **Search** and **Filter** fields to limit the attributes displayed for the dimension table.
9. Select the attributes that you want to use from the dimension table and indicate an attribute to be used as the key for joining with a fact table in the target subject area.
10. If any of the selected attributes have been removed or modified in the source table since the last refresh, then you see such columns highlighted and a message asking whether you want to update the table. Select **OK** in the message to reload the source columns. If you want to review the changes to the source columns, then click **Cancel** in the message, and later click **Refresh** to reload the source columns. If any of the attributes that you haven't selected have been removed or modified in the source table, then you see the refreshed list of source columns. If any of the custom columns fail validation during the refresh, then you see a message asking you to resolve the cause of failure and revalidate.
11. Optional: Click **Create Column** to add another column to your dimension table in the target subject area using these instructions:
    a. In Create Column, enter a display name.
    b. Under **Data Elements**, search for a data element from the physical table of the selected dimension table.
    c. From the search results, double-click the data element to place it in the text pane.
    d. Under **Functions**, search for a function to construct a column using expressions. For example, search for functions like "substring" or "concatenate" to construct new expression-based columns. From the search results, double-click the applicable result to add it to the central text pane.
    e. Click **Validate**, and then click **Save**.

12. In step 3 of the wizard, select, drag, and drop available data elements into the Selected Data elements pane to design a hierarchy for the dimension and then click **Next**. In the Selected Data Elements pane, click a level to update its primary key and set its display attribute in the Properties pane.

    You can add multiple levels in your hierarchy by right-clicking at a level and selecting Add Child or Add ‘n’ Child Levels. For example, your Region Hierarchy can have Region Total at Level 1, Region at Level 2, Country at Level 3, State at Level 4, and City at Level 5.

13. In step 4 of the wizard, select **Skip Joins** if you don’t want to join the selected dimension table to any facts. To join the selected dimension table to a fact, select the fact table, fact key, and join type. Click **Content Level** to specify the content level for your fact.

    You can join a single fact key column to multiple dimension keys.

    > **Note:**
    >
    > Ensure that the data types of the join key pairs match. If your data types don't match but you want to proceed, then click Yes in the message. However, if the data types can't be absolutely matched, then the server-side validation rejects that join completely and you must change the data type of custom key column to match the factory data type.

14. Optional: Click **Add Fact Table** to select another fact table to link your dimension to and define the join.

15. Click **Next**.

16. Optional: In step 5 of the wizard, select the subject areas to include the new dimension and click **Finish**.

You see a message that your step is being applied to the customization branch. After it's applied, you see the new customization step in the customization branch. You can now apply the customization branch to the main branch or edit it to add more steps.

> **Note:**
>
> If you’ve created Add a Dimension steps using the previous functionality, you can still edit and reapply through the Edit option.

Add a Fact Table

Add a fact table to an existing subject area.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.
3. On the Semantic Model Extensions page, click **User Extensions**.

   You see the main and existing customization branches.
4. In the User Extensions region, under Customization Branches, click a branch to open the Branch page.
5. On the Branch page, click **Add Step**.
6. In Add Step, select **Add a Fact**.

You see the wizard sequence to add a fact.

7. In step 1 of the wizard, enter a name for your customization step, for example, `Add Travel Expense` and add a brief description.
8. Select a target subject area to which you want to add the fact. For example, `Financials - AP Expense`.

   You see the details of the selected subject area.
9. Click **Next**.
10. In step 2 of the wizard, select the schema, and then select a view or table or synonym as the object. For example, `FCT_CALC_Extensions`.

    You see the fact table for the selected object.
11. In the details of the fact table for the selected source table, click the **Select Fact** and **Use for Key** check boxes for the source columns that you want to add to your new fact table in the target subject area.
12. Optional: In the details of the fact table for the selected source table, under **Select Degen Attribute**, click the check boxes for the attributes for which you need the degenerate dimension to be created.
13. If any of the selected attributes have been removed or modified in the source table since the last refresh, then you see such columns highlighted and a message asking whether you want to update the table. Select **OK** in the message to reload the source columns. If you want to review the changes to the source columns, then click **Cancel** in the message, and later click **Refresh** to reload the source columns. If any of the attributes that you haven't selected have been removed or modified in the source table, then you see the refreshed list of source columns. If any of the custom columns fail validation during the refresh, then you see a message asking you to resolve the cause of failure and revalidate.
14. Optional: Click **Create Column** to add a new column to your new fact table in the target subject area using these instructions:
    a. In Create Column, enter a display name.
    b. Under **Data Elements**, search for a data element from the physical table of the selected dimension table.
    c. From the search results, double-click the data element to place it in the text pane.
    d. Under **Functions**, search for a function to construct a column using expressions. For example, search for functions like "substring" or "concatenate" to construct new expression-based columns. From the search results, double-click the applicable result to add it to the central text pane.
    e. Click **Validate**, and then click **Save**.
15. Click **Next**.
16. In step 3 of the wizard, use the Diagram or Tabular tabs to specify the joins to link your new fact table to the dimensions in the selected subject area.

    Follow these instructions to specify the joins using the Diagram tab:
    a. Click **Add Table**.
    b. In Add Table, select the dimensions to add and click **OK**.
    c. Drag from the dimension's port (dark green circle) to the fact table's port (brown circle) to create a join link. You see the Join dialog.
    d. In the Join dialog, select the type of join, the fact table column, and the dimension key column. Click **Add Join Condition** to add multiple join conditions and then click **Join**.

If you want to provide expressions as join conditions, then click **Complex Join** and in Create Joins, click **Add Joins**, select the target and source logical tables, enter the join condition as an expression, and click **OK**.

Follow these instructions to specify the joins using the Tabular tab:

a. Click **Add Join**.

b. In Add Table, select the dimension to add and click **OK**. You see the Join dialog.

c. In the Join dialog, select the type of join, the fact table column, and the dimension key column. Click **Add Join Condition** to add multiple join conditions and then click **Join**. If you want to provide expressions as join conditions, then click **Complex Join** and in Create Joins, click **Add Joins**, select the target and source logical tables, enter the join condition as an expression, and click **OK**.

17. Optional: Click **Skip Joins** if you don't want to join a dimension now.
18. Click **Next**.
19. In step 4 of the wizard, select the aggregation rule for each fact column to set the aggregation behaviors.
20. Optional: You can set the time-balanced aggregation rule for a time dimension and hierarchy level-based aggregation rule for a dimension using these steps:

    a. For a fact column, click the Time-Balanced Aggregation icon.

    b. In the Time-Balanced Aggregation dialog, click **Add Time Dimension**, adjust the aggregation rule, and then click **OK**.

    c. For a fact column, click the Hierarchy Level-Based Aggregation icon, select the dimension and level. Click **Add Dimension** to add more dimensions. Click **OK**.

    Use a time-balanced aggregation when the added measure mustn't be "aggregated" by default across a time dimension. Oracle NetSuite Analytics Warehouse supports non-aggregation types like "Last" or "First" in place of the "SUM" aggregation type when required. Use a level-based aggregation when the underlying measure must always be calculated to a specific level of a predefined dimensional hierarchy. For example, in a product hierarchy that has the Product Total, Product Category, Product Sub-Category, and Product Details levels, you add a new measure called "Revenue" and need this "Product Category Revenue" measure to be aggregated to Product Category, then you must use the level-based aggregation and choose the right level of the Product Dimension. This setting enables Oracle NetSuite Analytics Warehouse to always aggregate and show the value of the measure at the Product Category level. This is useful when you need to calculate Product Revenue as a % of Category Revenue.

21. Click **Next**.
22. Optional: Select additional subject areas to add the fact.
23. Click **Finish**.

    You see a message that your step is being applied to the customization branch. After it's applied, you see the new customization step in the customization branch. You can now apply the customization branch to the main branch or edit it to add more steps.

Add a Hierarchy

Add a hierarchy to a dimension table in an existing subject area.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.

3. On the Semantic Model Extensions page, click **User Extensions**.

   You see the main and existing customization branches.

4. In the User Extensions region, under Customization Branches, click a branch to open the Branch page.

5. On the Branch page, click **Add Step**.

6. In Add Step, select **Add a Hierarchy**.

   You see the wizard sequence to add a hierarchy.

7. In step 1 of the wizard, enter a name for your customization step, for example, `Add Region Hierarchy` and add a brief description.

8. Select the subject area, the folder of the dimension table, and the dimension table to which you want to add a hierarchy.

   You see the existing hierarchies and the hierarchy levels in the selected dimension. If there aren't any hierarchies, then you see a message informing you that there are no hierarchies in the selected dimension.

9. Click **Next**.

10. In step 2 of the wizard, assemble the product hierarchy using the attributes from this dimension table with these instructions:

    a. Enter a hierarchy name. For example, Region Hierarchy.

    b. Select, drag, and drop available data elements into the Selected Data elements pane to design a hierarchy that you want. You can add multiple levels in your hierarchy by right-clicking at a level and selecting **Add Child** or **Add 'n' Child Levels**. For example, your Region Hierarchy can have Region Total at Level 1, Region at Level 2, Country at Level 3, State at Level 4, and City at Level 5.

    c. In the Selected Data Elements pane, click a level to update its primary key and set its display attribute in the Properties pane.

11. Click **Next**.

12. Optional: In step 3 of the wizard, select additional subject areas to include the new hierarchy.

13. Click **Finish**.

    You see a message that your step is being applied to the customization branch. After it's applied, you see the new customization step in the customization branch. You can now apply the customization branch to the main branch or edit it to add more steps.

Add Session Variables

Add custom session variables that you can include in the analyses. After merging this step to the Main branch and publishing it, the custom session variables are available in the custom security configuration user interface only.

The SQL query that you define is executed by user OAX$OAC. If you're using another schema in the query, then you must mention the schema name as prefix. You must ensure to grant user OAX$OAC access to all the database objects used in the query.

1. Sign in to your service.

2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.

3. On the Semantic Model Extensions page, click **User Extensions**.

You see the main and existing customization branches.

4. In the User Extensions region, under Customization Branches, click a branch to open the Branch page.
5. On the Branch page, click **Add Step**.
6. In Add Step, select **Add Session Variables**.

   You see the wizard sequence to add the session variables and a list of existing session variables.
7. In step 1 of the wizard, check if any of the existing session variables serve your purpose.

   If yes, then you can exit the wizard and use the applicable existing session variables in your analyses. If no, then continue with the next steps to create the session variables that you require.
8. Enter a name for your customization step, for example, `Add a Session variable using Invoice Received Date` and add a brief description. Click **Row-wise Initialization** to configure cache settings.
9. Click **Next**.
10. In step 2 of the wizard, define the SQL query and create the initialization block using these instructions:
    a. Enter a name and description for the initialization block.
    b. Select a preceding initialization block.
    c. Enter the SQL query that would be executed in the autonomous data warehouse and return a value that you can use in the reports. For example, if you want to get the Exchange Rate Type that's defined in Oracle NetSuite Analytics Warehouse into a session variable, then you can use the following SQL script:

    ```
    SELECT PARAMETER_VALUE FROM DW_CONTENT_PARAM_CONFIG WHERE
     PARAMETER_CODE='PARAM_GLOBAL_EXCHANGE_RATE_TYPE'
    ```
11. Click **Next**.
12. In step 3 of the wizard, create the session variables using the output of the initialization block created in step 2 of the wizard.
13. Click **Finish**.

    You see a message that your step is being applied to the customization branch. After it's applied, you see the new customization step in the customization branch. You can now apply the customization branch to the main branch or edit it to add more steps.

Extend a Dimension

Extend prebuilt dimensions with additional attributes from another data source.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.
3. On the Semantic Model Extensions page, click **User Extensions**.

   You see the main and existing customization branches.
4. In the User Extensions region, under Customization Branches, click a branch to open the Branch page.

5. On the Branch page, click **Add Step**.
6. In Add Step, select **Extend a Dimension**.

   You see the wizard sequence to extend a dimension.
7. Enter a name for your customization step, for example, `Add Cost Center Type`.
8. Select a target subject area, for example, `Financials GL Profitability`.
9. In **Folder**, select a dimension that you want to extend, for example, `Cost Center`.
10. Select a logical table, for example, `Dim – Cost Center`.

    You see the available attributes in the table.
11. Click **Next**.
12. Select a schema and table from the database.

    You see the available attributes in the table.
13. Select the columns that you want to expose or use as a key for creating the join.
14. Click in the **Display Name** table field to enter a new name for the column or to edit an existing one and then click **Enter** to accept or click **Esc** to cancel.
15. If any of the selected attributes have been removed or modified in the source table since the last refresh, then you see such columns highlighted and a message asking whether you want to update the table. Select **OK** in the message to reload the source columns. If you want to review the changes to the source columns, then click **Cancel** in the message, and later click **Refresh** to reload the source columns. If any of the attributes that you haven't selected have been removed or modified in the source table, then you see the refreshed list of source columns. If any of the custom columns fail validation during the refresh, then you see a message asking you to resolve the cause of failure and revalidate.
16. Optional: Click **Create Column** to create a new column in the selected dimension table using these instructions:
    a. In Create a new column, enter a display name, for example, `Cost Type`.
    b. Under **Data Elements**, search for a data element from the physical table of the selected dimension table.
    c. From the search results, double-click the data element to place it in the text pane.
    d. Under **Functions**, search for a function to construct a column using expressions. For example, search for functions like "case" to construct new expression-based columns. From the search results, double-click the applicable result to add it to the text pane.
    e. Click **Validate**, and then click **Save**.
17. Click **Save**.
18. Click in the **Source Column** table field to edit the column definition.
19. Click **Save**.

    You see the new column in the Data preview section in a highlighted color.
20. Click **Next**.
21. Select a join key to pair with the source column. If you want to provide expressions as join conditions, then click **Complex Join** and in Create Joins, click **Add Joins**, select the target and source logical tables, enter the join condition as an expression, and click **OK**.
22. Click **Next**.
23. Select the subject areas that should use this customization.

> **Note:**
>
> The Subject Area that you initially selected is selected by default and is read-only. By default, all additional subject areas are selected. Deselect the additional subject areas that shouldn't use this customization.

24. Click **Finish**.

    You see the new customization step in the customization branch. You can now apply the customization branch to the main branch or edit it to add more steps.

Add Derived Columns

Add a derived column to an existing subject area.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.
3. On the Semantic Model Extensions page, click **User Extensions**.

   You see the main and existing customization branches.
4. In the User Extensions region, under Customization Branches, click a branch to open the Branch page.
5. On the Branch page, click **Add Step**.
6. In Add Step, select **Add a Column**.

   You see the wizard sequence to add a column.
7. In step 1 of the wizard, enter a name for your customization step, for example, `Regional Revenue` and add a brief description.
8. Select a target subject area to which you want to add the column. For example, `Profitability`.

   You see the details of the selected subject area.
9. Select the presentation folder within the selected subject area and the logical table to which you want to add the column.
10. Click **Next**.

    You see the Create Column dialog in step 2 of the wizard.
11. In step 2 of the wizard, define your new column using these instructions:
    a. In Create Column, enter a display name.
    b. Under **Data Elements**, search for a data element from the subject area that you had selected previously.
    c. From the search results, double-click the data element to place it in the text pane.
    d. Under **Functions**, search for a function to construct a column using expressions. From the search results, double-click the applicable result to add it to the central text pane. For example, search for functions like "Filter" or "Avg" to construct expression-based columns. A sample expressions to derive the average supplier payment days is `avg(ROUND(((CASE WHEN Invoice Received Date is not null THEN (Financials - AP Payments.Payment Date.Payment Date - Invoice Received Date) ELSE (Financials - AP Payments.Payment Date.Payment Date - Financials - AP`

```
Invoices.Invoice Date.Invoiced Date) END)/Financials - AP Payments.Facts -
Analytics Currency.Total Payment Count),0)).
```

e. Click **Validate**, and then click **Save**.

12. Optional: If you want the underlying measure of the column to be calculated to a specific level of a predefined dimensional hierarchy, then complete these steps:
    a. Click the **Hierarchy Level-Based Aggregation** icon.
    b. In the Hierarchy Level-Based Aggregation dialog, select the dimension, level, and then click **OK**.
    c. Click **Add Dimension** to add more dimensions.
13. Click **Next**.
14. Optional: Select additional subject areas to add the fact.
15. Click **Finish**.

    You see a message that your step is being applied to the customization branch. After it's applied, you see the new customization step in the customization branch. You can now apply the customization branch to the main branch or edit it to add more steps.

Create a Subject Area

You can create a subject area as a container and later add dimensions and facts to your new subject area or create a subject area based on an existing one.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.
3. On the Semantic Model Extensions page, click **User Extensions**.

   You see the main and existing customization branches.
4. In the User Extensions region, under Customization Branches, click a branch to open the Branch page.
5. On the Branch page, click **Add Step**.
6. In Add Step, select **Create a Subject Area**.

   You see the wizard sequence to create a subject area.
7. In step 1 of the wizard, enter a name for your customization step, for example, `Custom Profitability` and add a brief description.
8. Create a subject area using one of the methods:
   a. Select **Create a Subject Area** to create a subject area container, provide a name without any leading or trailing white spaces, add a description, and then click **Next**. You see step 4 of the wizard. Click **Finish** to create the subject area.
   b. Select **Create a Subject Area based on an existing one** to create a subject area using an existing one in the system and provide these details:
      i. Select an existing subject area, name your subject area, and then click **Next**.
      ii. In step 2 of the wizard, select the data elements that you want in your new subject area.
      iii. Click **Add Subject Area** to select and add data elements from multiple subject areas.

iv. In step 3 of the wizard, organize and rename the data elements in your new subject area.

v. Click **Next**.

vi. In step 4 of the wizard, review your new subject area and click **Finish** to create it.

You see a message that your step is being applied to the customization branch. After it's applied, you see the new customization step in the customization branch. You can now merge the customization branch with the main branch or edit it to add more steps.

Modify a Subject Area

Modify a custom subject area using these instructions.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.
3. On the Semantic Model Extensions page, click **User Extensions**.

   You see the main and existing customization branches.
4. In the User Extensions region, under Customization Branches, click a branch to open the Branch page.
5. On the Branch page, click **Add Step**.
6. In Add Step, select **Modify a Subject Area**.

   You see the wizard sequence to modify a subject area.
7. In step 1 of the wizard, enter a name for your customization step, for example, `Custom AP Invoices` and add a brief description.
8. Select a subject area that you had created using the instructions in Create a Subject Area.

   You see the details of the selected subject area.
9. Click **Next**.
10. In step 2 of the wizard, from the Available Data Elements pane, select or deselect the data elements that you want to use or don't want in the selected subject area.
11. Optional: Click **Add Subject Area** to select and add data elements from multiple subject areas.
12. Click **Next**.
13. In step 3 of the wizard, organize and rename the data elements in your modified subject area.
14. Optional: Click the **Advanced Properties** icon next to the custom subject area to select an implicit fact that allows dimensions to be used for analytic queries even when not joined to a logical fact table.
15. Click **Next**.
16. In step 4 of the wizard, review your modified subject area and click **Finish**.

Edit or Delete a Branch Step

As the owner of the branch or an user with the Modeler Administrator role, you can edit a step to modify the details or delete it if it's no longer required. You can delete multiple steps together.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.
3. On the Semantic Model Extensions page, click **User Extensions**.
4. In the User extensions region, under Customization Branches, click a branch to display the steps.
5. Hover over a step to view **Actions**.
6. Click **Edit** and update the details.
7. Click **Delete** to remove it from the branch.

Reapply a Branch Step

As the owner of a branch or a user with the Modeler Administrator role, you can reapply a failed step after resolving any issues that might have occurred with tables or columns in the autonomous data warehouse.

You can reapply the steps from the Branch details page without opening the steps.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.
3. On the Semantic Model Extensions page, click **User Extensions**.

   You see the main and existing customization branches.
4. In the User Extensions region, under Customization Branches, click a branch to display the steps.
5. Hover over a step to view **Actions**.
6. Click **Reapply**.

Disable and Enable the Disabled Steps

You can disable and enable the disabled steps in the main and customization branches in the development and production environments. This helps you in troubleshooting issues in the branches.

You can perform these actions on the steps in the User Extensions and Security Configurations regions. You can disable and enable individual steps or select multiple steps using the **Manage Steps**, **Enable**, and **Disable** buttons on the branch details page. You can't disable a published step.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.
3. On the Semantic Model Extensions page, click **User Extensions**.

   You see the main and existing customization branches.
4. In the User Extensions region, click **Main** to view the details.
5. On the Main branch page, hover over a step to display the options.
6. Click **Disable**.

7. In Confirm Disable Steps, click **Disable**.
8. On the Main branch page, click **Manage Steps**, select the check box for the disabled steps, and then click **Enable**.
9. In Confirm Enable Steps, click **Enable**.
10. Perform these actions for steps in the customization branches.

Copy Steps from One Branch to Another

Use these instructions to copy steps from a customization branch to another customization branch.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.
3. On the Semantic Model Extensions page, click **User Extensions**.

   You see the main and existing customization branches.
4. In the User extensions region, under Customization Branches, click a branch whose steps you want to copy.
5. On the Branch page, click **Manage Steps**.
6. Select the check box for the steps that you want to copy and click **Copy**.
7. In Copy Steps, select the target branch to which you want to copy the selected steps and click **OK**.

   Alternately, you can create a branch using the "Create Branch" option available in the Copy Steps dialog and copy the steps into this new branch in a single action.

   You see a confirmation message that the steps have been successfully copied.

View Details of Failed Branch Steps

You can view the reasons why a branch step had failed and then correct the errors.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.
3. On the Semantic Model Extensions page, click **User Extensions**.

   You see the main and existing customization branches.
4. In the User extensions region, under Customization Branches, click the branch with the **Failed** icon to view the details.
5. On the Branch page, click the **Failed** status for a step that has failed and view the error details.

Merge the Customization Branches with the Main Branch

Merge the customization branches with the main branch to use the customization steps as the sequence of steps.

When you edit a branch, the system automatically locks it to prevent another user from simultaneously editing the same branch. As you complete each step or reorder the steps, the system unlocks the branch, saves it, and notes the change in the change log tab.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.
3. On the Semantic Model Extensions page, click **User Extensions**.

   You see the main and existing customization branches.
4. In the User Extensions region, under Customization Branches, hover over a branch to view **Actions**.
5. From **Actions**, click **Merge to Main Branch**, and then click **Ok**.

   When a branch gets merged into Main, the other branches go out of synchronization and you must resynchronize them with the Main branch.

Reorder Steps of Customization Branches

You can reorder the steps in a branch that has been applied or is yet to be applied to the main branch.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.
3. On the Semantic Model Extensions page, click **User Extensions**.

   You see the main and existing customization branches.
4. In the User Extensions region, under Customization Branches, click a branch to display the steps.
5. On the Branch page, click **Manage Steps**.
6. Select the check box for the steps that you want to reorder and click **Reorder**.
7. In Reorder Steps, use the drag handles to drag and drop the steps in the new order that you want, and then click **Reorder**.

Delete a Main Branch Step

As a modeler administrator, you can delete all steps of a main branch that have been either applied or have failed. With modeler permissions, you can delete only failed steps of a main branch.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.
3. On the Semantic Model Extensions page, click **User Extensions**.

4. In the User Extensions region, click **Main** to view the details.
5. On the Main Branch page, hover over a step to display the options.
6. Click **Delete** to remove it from the main branch.

Tag the Main Branch's Steps

You can create tags on the "Applied" steps of the Main branch as a snapshot at a given point in time.

When you have a set of customizations ready for promotion and merged them with the Main branch, you can tag the Main branch's steps with “Applied” status using the **Create Tag** option for the Main branch or tag any of it's steps with “Applied” status directly on the Main branch detail page using the **Tag** option. When you tag a step directly on the Main branch details page or include a step while tagging using the **Create Tag** option for the Main branch, the steps prior to the selected step are included in the tag too. You can untag the tags that you create for the Main branch's steps using the **Untag** option.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.
3. On the Semantic Model Extensions page, click **User Extensions**.

   You see the main and existing customization branches.
4. To create a tag for a step in the Main branch, in the User Extensions region, click **Create Tag**.
5. In Create a Tag, enter a name for the tag and a description.
6. Select a merged step that you want to include in this tag.
7. Click **Done**.
8. To tag a step directly, hover over the Main branch to view **Actions** and then click **View Details**.
9. On the Main branch details page, hover over a step to view **Actions** and then click **Tag**.
10. In Create a Tag, enter a name for the tag and a description.
11. Click **Done**.
12. To untag a tag for the Main branch and the step included in the Tag, on the Main branch details page, hover over a step to view **Actions** and then click **Untag**.
13. In Confirm Untag, click **Untag**.

> **Note:**
>
> If there're multiple tags on the same step, then you can select the check boxes for the applicable tags in the Confirm Untag dialog.

Publish the Model

You can publish the versions on the main development branch and the other branches in the non-production environments such as development or test to ensure that there are no errors.

While publishing the data model, you can select the user extensions and security configurations that you added as part of customizing the semantic model. If you select the security configurations, then Oracle NetSuite Analytics Warehouse applies them on the user extensions that you selected. If the security configurations refer to elements in the model that aren't part of the user extensions, then Oracle NetSuite Analytics Warehouse excludes them at the time of publishing the model.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.
3. On the Semantic model Extensions page, click **Publish Model**.
4. In Publish Model, select the user extensions and security configurations that you want to publish.
5. Click **Publish**.

Load Customization Data to the Autonomous Data Warehouse

You can load your customization data to the autonomous data warehouse provisioned with your Oracle NetSuite Analytics Warehouse instance.

You need the autonomous data warehouse wallet and credentials of the administrator.

1. Connect to the autonomous data warehouse corresponding to your Oracle NetSuite Analytics Warehouse instance.
2. Create a custom schema to store the customization data.

   Syntax: `create user <custom_schema-name> identified by <custom_schema-password>;`

   Example: `create user example_schema identified by abcDEF123654;`
3. Create one or more tables in the custom schema that you created.

   Syntax: `CREATE TABLE <custom_schema-name>.<custom_extent_table_name> (<parameters>));`

   Example: `CREATE TABLE example_schema.ABC_EXTN ("DATE" DATE, "CATEGORY" VARCHAR2(1024 BYTE) , "MANAGER" VARCHAR2(1024 BYTE) );`
4. Populate the required data and grant select permissions to the OAX$OAC schema in the autonomous data warehouse using this script:

   Syntax: `grant select on <custom_schema-name> to OAX$OAC;`

   Example: `grant select on example_schema.ABC_EXTN to OAX$OAC;`
5. Commit the changes to the autonomous data warehouse and disconnect.

Add Security Configurations

Add security configurations to secure the subject areas and data with prebuilt and custom duty and data type of application roles.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.
3. On the Semantic Model Extensions page, click **Security Configurations**.
4. In the Security Configurations region, click **Add Configure Data Security** to secure your data with the data type of application roles.

   See Configure Data Security.
5. In the Security Configurations region, search for the prebuilt "Configure Object Permissions" to configure permissions for objects such as subjects areas and their elements with duty type of application roles.

   See Configure Object Permissions.
6. Optional: In the Security Configurations region, click **Reapply Steps** to validate the security configuration-related steps against the current state of the model.

Configure Data Security

As a security administrator, provide users with access to data using the custom-created data type application roles.

You can add filters to data retrieved from logical or presentation objects based on the data roles assigned to users. If you specify functional groups, Oracle Fusion Data Intelligence combines all the filters in the same functional group using the OR operator and combines all sets of filters in different functional groups using the AND operator.

You can add one customization step for each data role. The elements that you can secure are from the Main branch of the semantic model. Hence, if you need a newly added object to be secured, then you must ensure that the customization branch containing the newly added object is merged with the Main branch before configuring the security. If any of the custom-created role is no longer available, then the security configuration for that role is removed from the "Configure Data Security" step.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.
3. On the Semantic Model Extensions page, click **Security Configurations**.

   You see the existing security configurations, if any.
4. In the Security Configurations region, click **Add Configure Data Security Step**.
5. In step 1 of the wizard, enter a name for your step, select a data type application role, and then click **Next**.
6. In step 2 of the wizard, from the **Available Objects** drop-down list, click either Presentation Objects or Logical Objects to select the objects that you want to secure with the selected data type application role.

   If you're viewing the presentation objects, then expand the subject area folders and double-click the objects. If you're viewing the logical objects, then double-click the logical table folders or expand the table folders and double-click the objects. You see the selected objects under **Object to be secured** in the right pane.
7. Optional: In **Functional Group**, select a prebuilt functional group or enter a unique functional group name in the text box to create a custom functional group.
8. For the **Filter Argument**, click the **Function** icon to define how the data filter gets applied.

Use the Expression Editor to enter the filter, based on the session variables that you had created previously. To view an example, see Custom Security in Fusion Data Intelligence.

9. Click **Next**.
10. Click **Finish**.
11. Optional: In the Security Configurations region, click **Reapply Steps** to validate the security configuration-related steps against the current state of the model.

Configure Object Permissions

Configure the permissions for objects such as subject areas and its elements with the ready-to-use or the custom-created duty roles.

You secure the subject areas and their elements using the **Configure Object Permissions**, a prebuilt single step. You edit this single step to specify the subject areas, their elements, and the duty roles to secure these with. The elements that you can secure are from the Main branch. Hence, if you need a newly added object to be secured, then you must ensure that the branch containing the newly added object is merged with the Main branch before configuring the security. If a custom-created role is no longer available, then the security configuration for that role is automatically updated in the existing Configure Object Permissions step.
For the front-end objects such as key metrics and workbooks, set the permissions individually for each object by adding the applicable duty role and the corresponding access.

By default, the list of permissions by duty role displays the explicit permissions set for the subject area or the elements of the selected subject area. If you want to add more permissions, then select the duty role from the list and set the required permission. Permission levels that you can set are:

- Default (inherited from the parent element).
- No Access (deny access to the respective subject area or its elements)
- Read-only (access to read the respective subject area or its elements).

Repeat the operation for all the subject areas or the subject area elements that you need to secure.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.
3. On the Semantic Model Extensions page, click **Security Configurations**.

   You see your existing security configurations and the prebuilt object permissions-related step.
4. In the Security Configurations region, search for the prebuilt "Configure Object Permissions", and hover over it to view **Actions**, and then click **Edit**.
5. In step 1 of the wizard, click **Next**.
6. In step 2 of the wizard, select the subject areas or elements and set the corresponding desired permission to the duty role selected from the list, and then click **Next**.
7. Review your changes and click **Finish**.

View Activity History of Semantic Model Extensions

View an audit of all activities performed on the semantic model.

Amongst the activities, you may see that the system steps have been applied. Oracle NetSuite Analytics Warehouse applies the system steps in the following scenarios:

- Any changes to augmentations and new augmentations. A new augmentation replays all the steps in the Main branch because of implicit dependencies.
- New module activations. Module activations can run augmentations related to that module, but the Main branch steps are always replayed.
- Any changes done to flexfields (like DFFs) in your Oracle Fusion Cloud Applications instance such as deletion or addition. It doesn't matter where the DFF's are. If they are deleted in the source, then they get deleted in the prebuilt subject area. Hence, the custom subject area also must be modified to remove it. In this case, augmentation won't fail but the Main branch fails as it's custom and needs to be modified.
- Application update such as the Oracle NetSuite Analytics Warehouse version upgrade also runs the Apply System Steps.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.
3. On the Semantic Model Extensions page, click **Activity History** to view an audit of all activities performed on the semantic model.
4. Hover over an activity to view details of the activity.

Move Your Customizations to Another Environment

When you've a set of changes to data augmentations, custom attribute mapper, semantic model, security configurations, and customized KPIs that are ready for promotion, use the Bundles functionality to move the desired objects to other environments.

See Bundle Your Application Artifacts.

Republish Your Customizations

As the owner of the user extensions or a user with the Modeler Administrator role, you can republish them if you've modified them after publishing them initially.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Semantic Model Extensions** under **Application Administration**.
3. On the Semantic Model Extensions page, click **User Extensions**.
4. In the User Extensions region, you see a warning icon and the **Republish** button on the Semantic Model Extensions page.
5. Click **Republish** to publish your user extensions with the latest details.

Map the Custom Attributes

You can attach custom objects to transactions and transaction lines.

When you attach custom objects to transactions or transaction line types, you automatically create extension attributes that appear in Oracle Analytics Cloud reports and dashboards. The process is automated, so you don't need to create dimensions and extend entities.

Map Custom Attributes

Import custom attributes for transactions, transaction line, transaction accounting lines, and dimension entities using the Custom Attributes Mapper.

Determine the source attributes in NetSuite you want to import before starting this procedure to simplify the mapping process. For the source transaction types or entities that you want to import, NetSuite Analytics Warehouse displays the business names, IDs, data types, and size of the associated custom attributes.

> **Note:**
>
> Custom attributes that reference other transaction internal IDs aren't supported in the Custom Attributes Mapper. Use the Data Augmentation capability to bring these into NetSuite Analytics Warehouse.

The attributes that you can select have a default limit of 4000 characters for all data types. If the source transaction has attributes with more characters than the default limit, the system displays the unsupported attributes. If you want to use these unsupported attributes, you must either change the size in the data source or request the administrator to increase the character limit.

1. In NetSuite Analytics Warehouse **Console**, click **Data Configuration**.
2. On the Data Configuration page, under **Applications**, click **Custom Attributes Mapper**.
3. On the Custom Attribute Map List page, click **Create**.
4. Select the type of attribute you want to create, and the source transaction type or entity you want to import.
5. Select the attributes you want to associate with the source transaction type.
6. Click **Save**.
7. Click **Publish** to make the attribute map available to users.

5

Manage Oracle NetSuite Analytics Warehouse

As the cloud account administrator, manage the service instance for Oracle NetSuite Analytics Warehouse .

Topics:

About Application Updates

Application updates are available for major releases (quarterly basis), patches (monthly basis), and emergency fixes (as needed).

Except the emergency patches, you can decide when to apply the application updates. You can schedule the update for current and major releases to occur by a deadline. After that deadline, Oracle automatically updates your application. These updates have zero downtime.

The auto-update process updates the data model and then immediately after the data model update, this process updates the content. This process runs the incremental data pipelines as part of the data model upgrade process when upgrade is scheduled to run for the day. You can check the last refresh date either from the Console or the subject area to ensure that the data has been refreshed as part of the data model upgrade process.

To ensure a successful update, verify the connection between Oracle Fusion Data Intelligence and all of the sources is healthy. It's also required to define a data refresh schedule with a valid date, since the application update is triggered during the incremental load process.

You can view the application updates and plan accordingly using the Release updates tile under Service Administration on the Console. You also see a notification on the Data Configuration page when an application update is available. You can then plan to uptake using the Release Updates tile.The upgrade coincides with the daily incremental refresh schedule to optimize the process. This schedule uses Universal Coordinated Time (UTC), not your local time zone. Make sure to convert the intended upgrade time to UTC to prevent upgrade timing issues.

During the application upgrade, all records in the tables get updated with the date of the upgrade. The upgrade process then performs a full data refresh during which the data pipelines pull data from the tables based on the date of the upgrade. This process also performs an incremental data refresh. This upgrade adds new pillars, modules, data models (facts and dimension tables), key metrics, workbooks, and visualizations. The data pipelines for functional areas that have been activated are preserved during the upgrade. The data pipelines in Saved or Scheduled status are reset to Saved status. To activate these data pipelines, you must set them to Activated status. You can skip an upgrade for one release. However, you must upgrade when the next patch is available. The application upgrade adds new and replaces existing modified content with the latest version. If you've merged an

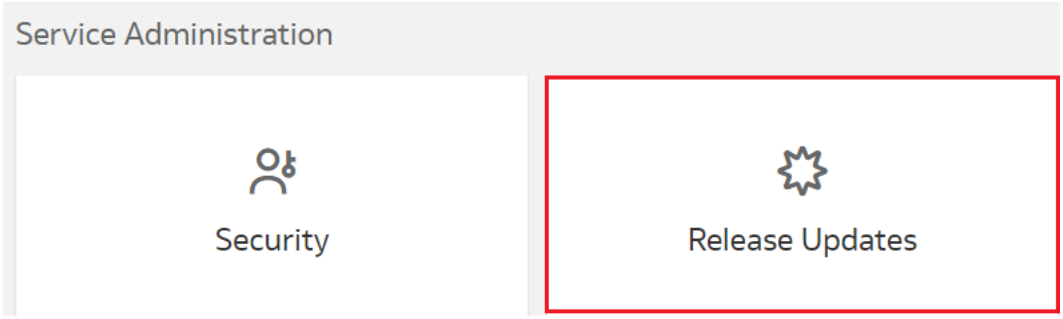
external application, then as part of the upgrade some objects within the external semantic model get upgraded too. Hence after every upgrade, you're recommended to download the latest semantic model uploaded through the user interface and make any changes on top of it.

Update Your Application

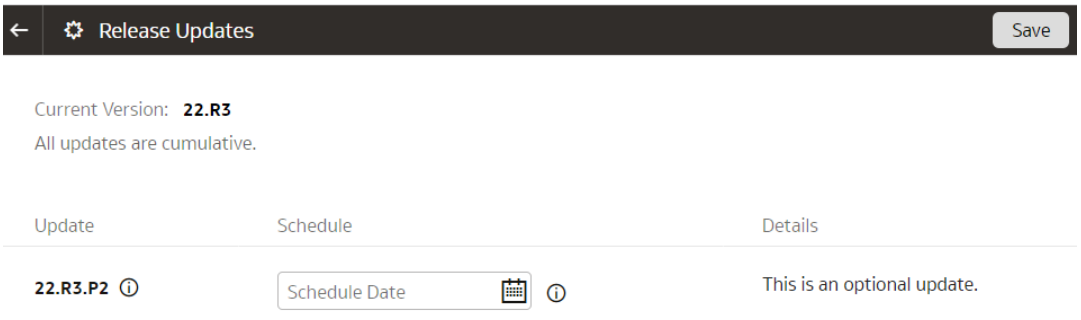
As the service administrator, you can view available application updates, schedule a date to accept the update, and know when an update isn't getting applied as expected.

Oracle NetSuite Analytics Warehouse provides automatic updates of the application to ensure you're using the most current software. You can schedule the update earlier than the predefined date and time. If you do nothing, the application upgrade automatically runs on the date shown on the Release Updates page. However, prior to updating your application, you must ensure that a data pipeline for at least one functional area has been activated.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Release Updates** under **Service Administration**.

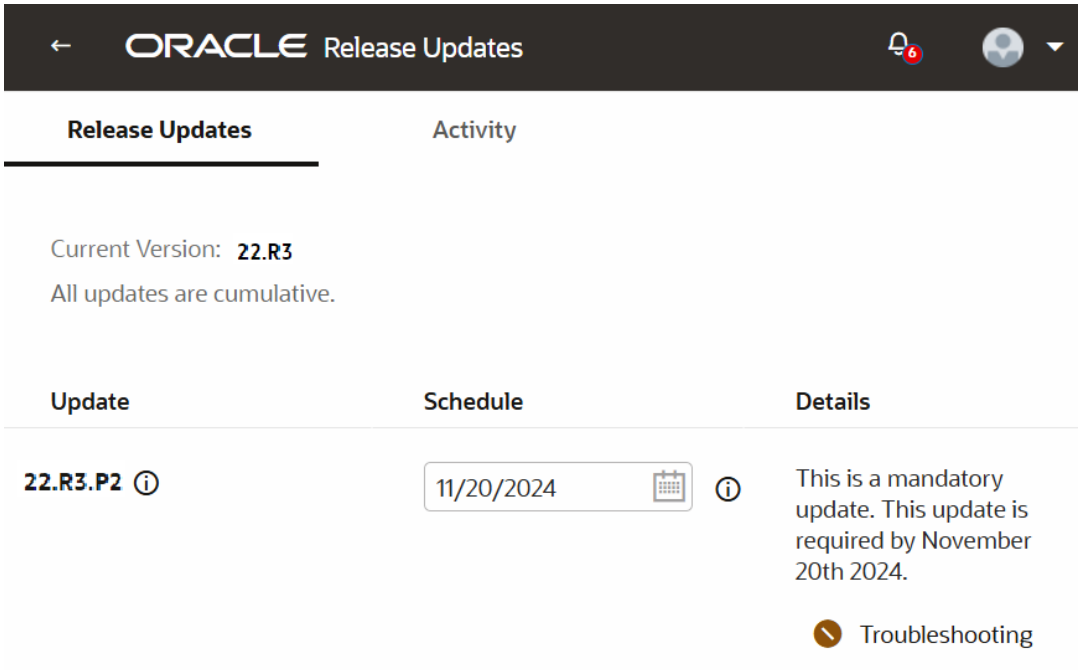


3. On the Release Updates page, select an application update under **Update**, and then in **Schedule**, select the date on which you want to apply it.



4. Click **Save**.
5. Verify that the update is applied as expected on the scheduled day.

   If there's an issue while applying the update, the system displays the **Troubleshooting** icon.



6. Click **Troubleshooting** to view the details and reach out for help to resolve the issue.

View Release Update Activity

You can view the actual date when a release was deployed for the instance and any service administrator actions such as date selections for release or patch uptake.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Release Updates** under **Service Administration**.
3. On the Release Updates page, click **Activity**.
4. In the Activity section, view the release update details.

Bundle Your Application Artifacts

As a service administrator, you can manage snapshots of your application artifacts as bundles.

Topics:

- View the Activity History of Bundles

About Bundles

Bundles are snapshots of your application artifacts such as configurations and customizations at a certain point in time.

Bundles work on environments where the source and target are at the same level or the source is at a lower version and the target is at a higher version. For example, a bundle generated from a previous content version works on the latest content version. Ensure that your bundle size is less than 1 GB. If the bundle size exceeds 1GB, then consider splitting the bundle by selectively including necessary artifacts. Prior to deployment, the system validates the bundles to confirm whether the bundles contain all prerequisites. If there is a validation error, then you see the applicable bundle with "Validation Failed" status and you can't deploy it. You must fix the issue and try to deploy again.

Bundles enable you to:

- Package custom development by defining a bundle that represents a subset of application artifacts in an environment such as development, test, or production.
- Migrate custom development and deploy the bundle on a target environment.
- Synchronize instances by promoting changes from one environment to another such as production to test.
- Restore the system when something goes wrong with an environment and you need to do a complete system restore.
- Create a backup of the environment or subset of application to save current state of the artifacts.
- Restore artifacts by importing from a bundle to restore state of the relevant artifacts to what was in the bundle.

You can bundle your application artifacts as:

- Data Config bundle: This includes pipeline parameters, activation metadata, data augmentations, and custom attribute mappings. You can install this bundle in an existing environment after a hard data reset. This is useful to leave content as-is and reset data pipeline. When you bundle data configurations, only the deployed data augmentations or configurations are included. Augmentations or configurations that aren't in deployed state in the source instance won't be included in a data configuration bundle.
- Semantic Model bundle: This includes main branch, tags, custom branch, and all the security customizations. Use this bundle to import the semantic model extensions because it allows you to select which tag and version to publish and what to publish.
- Security bundle: This includes custom application roles and custom data security.
- Content bundle: This includes snapshots of Oracle Analytics Cloud folders, projects, dataset definitions,workbooks, duty roles for content, and report parameters. The content bundle always merges the catalog content from source to target. While merging, if any conflicts are found, it replaces the content. It doesn't track the deleted content.
- Composite bundle: This includes one or more of the other bundles.
- Environment bundle: Environment bundle publishes the original semantic model without the customizations. This includes all artifacts of a specific environment to revert to a known state of system. For example, at the end of every week, the service administrator can create a bundle called DevEnv_YYMMDD to maintain a backup of the environment. You

must first deploy the data configuration bundle or manually activate your data pipelines before deploying the environment bundle.

> **Note:**
>
> Ensure that you've activated the functional areas and data is available prior to working with the semantic models or content. Either manually configure and activate your data pipelines in the target environment, or deploy a Data Config bundle to ensure that configurations and activations are at the same level as the source environment. Only then, it makes sense to deploy an Environment bundle, Semantic bundle, or Composite bundle because they depend on data.

Follow these recommended practices to ensure a smooth experience:

- Include the applicable security-related information in the Semantic Model and Content bundles.
- Reassign the groups to the users because the Security bundle doesn't overwrite the user-group mappings.
- Include the security configuration when you're exporting a Semantic Model bundle from a test to a production environment.
- While creating a Security bundle, if the number of application roles exceed 1000, then you may encounter an error. In such a case, use the **Select Application Roles** button to select specific roles.
- Use unique names for the semantic model extension steps. This enables the tags to work correctly while using the Content bundle to migrate your content to the target instance.

Create a Bundle

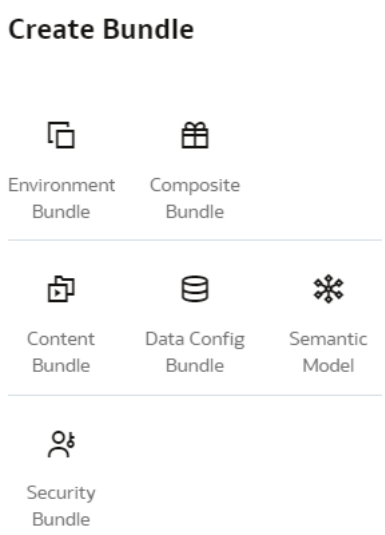
Create a snapshot of your application artifacts to save their current state. You can view the bundles that you created on the Bundles page.

While creating a bundle in your source instance, ensure that you select only the data sources that are also available in the target instance to avoid deployment failures in the target instance.

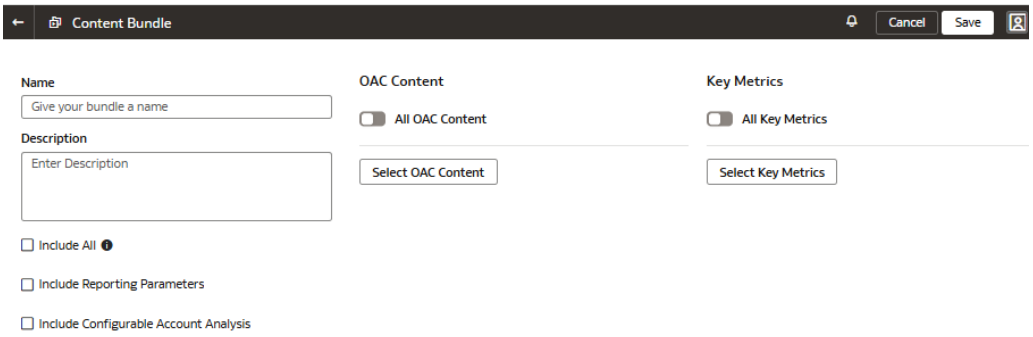
Prior to creating and generating a content bundle, ensure that none of the datasets have names with special characters. Also, if some of the datasets don't have key metrics to back up, manually select the key metrics from the respective subject areas and datasets and regenerate the content bundle. If there aren't any key metrics to back up in the bundle, then you can only select the **OAC Content** option from the user interface.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Bundles** under **Application Administration**.
3. On the Bundles page, click **Create**.

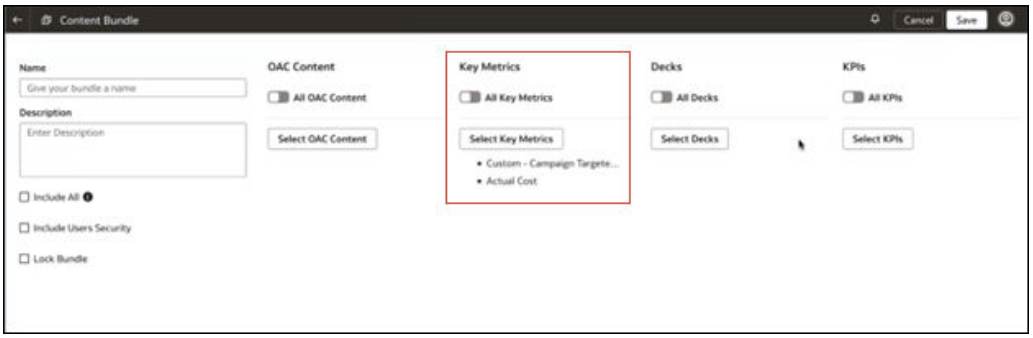
4. In Create Bundles, select the type of bundle that you want to create. For example, Content Bundle.



5. Enter a name and description for your bundle.
6. Depending on the type of bundle, select the applicable option and then click **Save**:
   - For a content bundle, select the **Include All Content** check box, or select the applicable Oracle Analytics Cloud content, workbooks, and key metrics using the corresponding toggles and buttons. You can select the **Include Reporting Parameters** and **Include Configurable Account Analysis** check boxes, if applicable.

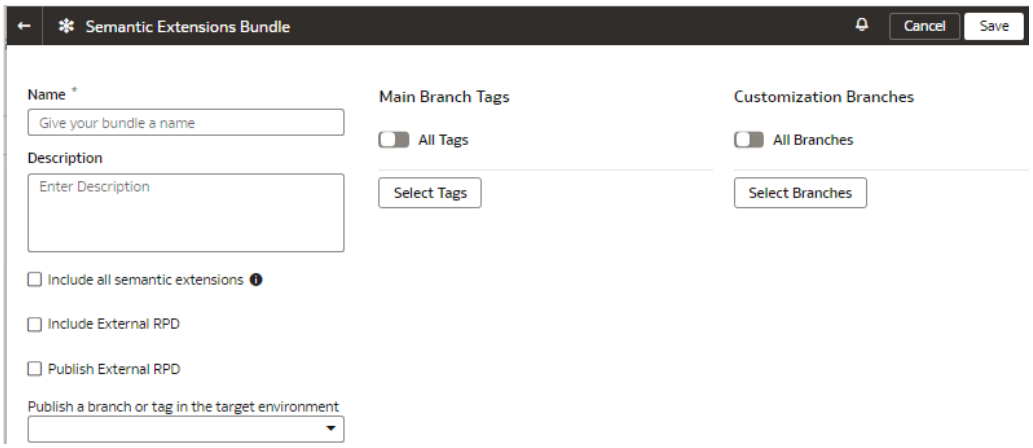


   - For a key metrics content bundle, select the key metrics you want to bundle using the corresponding toggles and buttons. Key metrics are also included with Environment Bundles. To improve performance, select the specific key metrics you want to bundle.

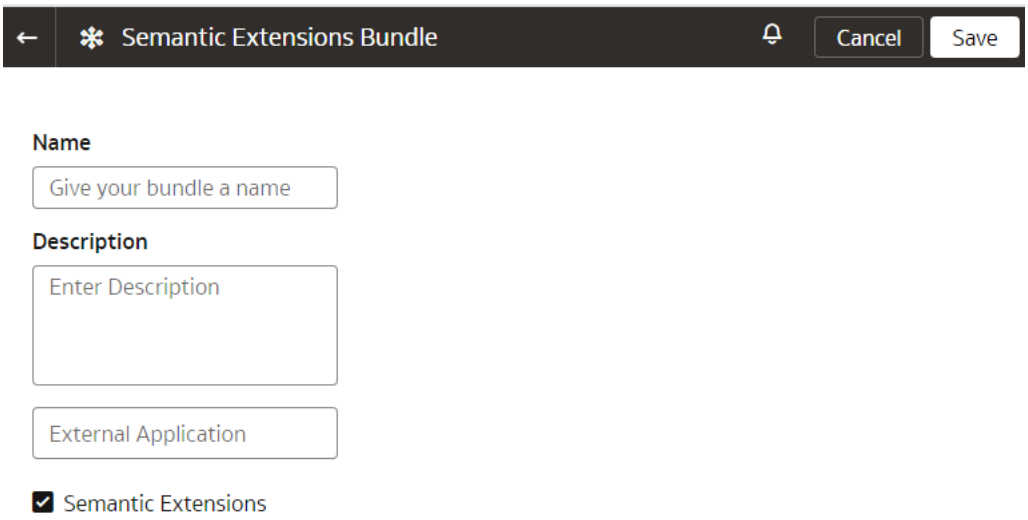


> **Note:**
>
> To import key metrics, you must deploy a key metric bundle into a new instance with key metrics enabled. You can import key metrics in datasets however the entire dataset won't be imported due to security limitations. Datasets without key metrics aren't included in key metrics.

- For a data configuration bundle, select the **Include all data configuration settings** check box, or select the applicable modules, augmentations, and custom attribute mappings using the corresponding toggles and buttons.
- For a semantic extensions bundle, select the **Include all semantic extensions** check box, or select the applicable branch or tag in the **Publish a branch or tag in the target environment** field, or select applicable tags and branches using the corresponding toggles and buttons. You can select the **Include External RPD** or **Publish External RPD** check boxes to include or publish the external semantic model.

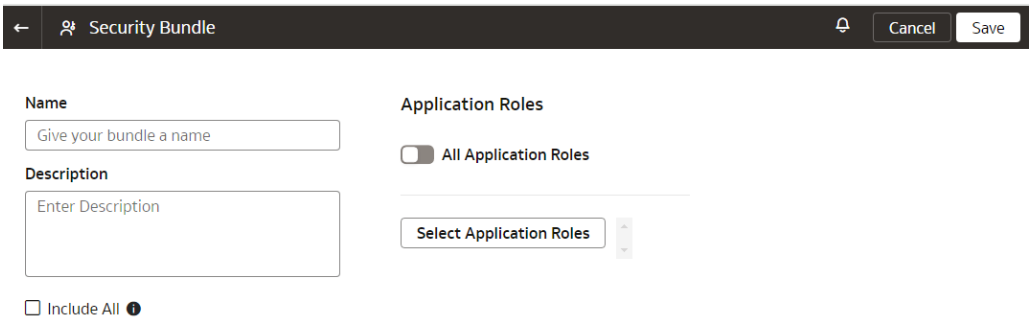


If you've migrated to the latest wizards for extending the semantic model, then select the applicable external application and select the **Semantic Extensions** check box to include all the extensions.

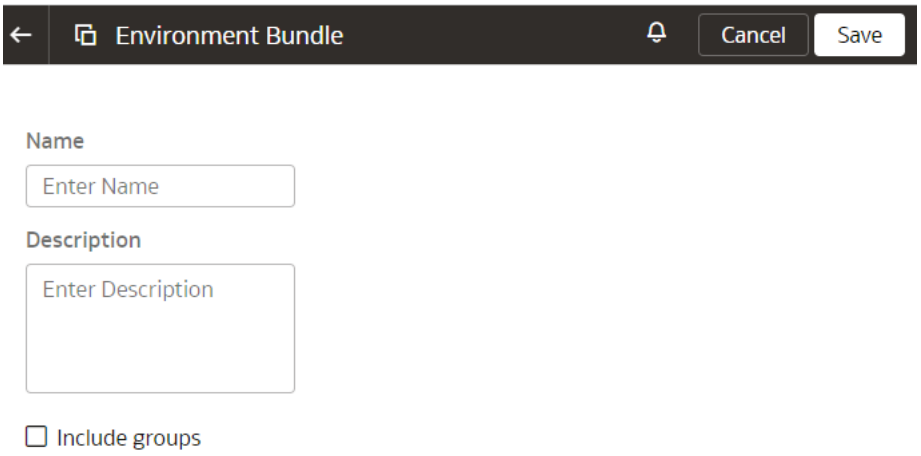


- For a security bundle, select the **Include all security settings** check box or select the Oracle NetSuite Analytics Warehouse-related application roles using the **Select Application Roles** button as the recommended approach.

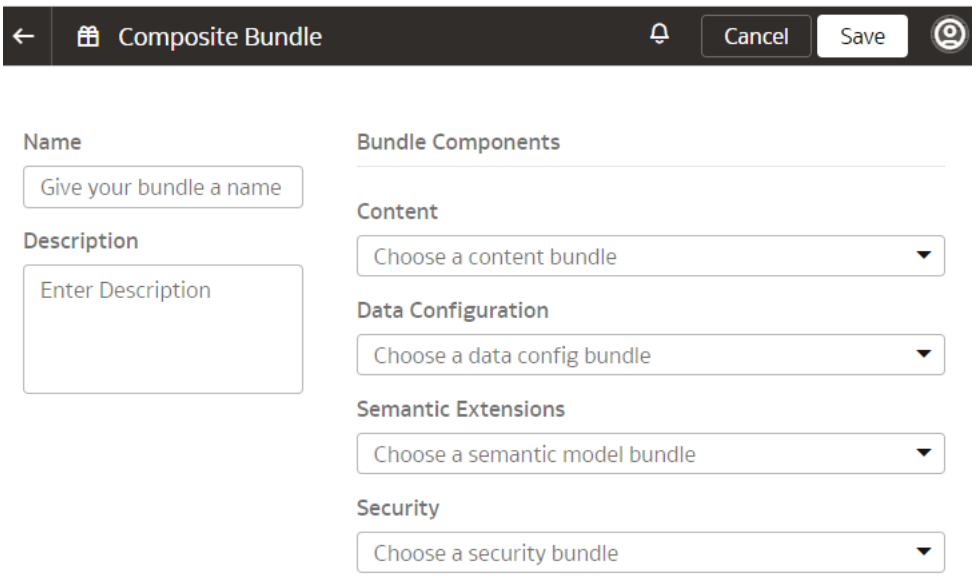
> **Note:**
>
> If you've uptaken the enhanced security capability of Oracle NetSuite Analytics Warehouse, then you no longer can specifically select groups as part of the Security bundle. See About Managing Users, Groups, and Application Roles.



- For an environment bundle, provide a name.



- For a composite bundle, select any of the other bundles.

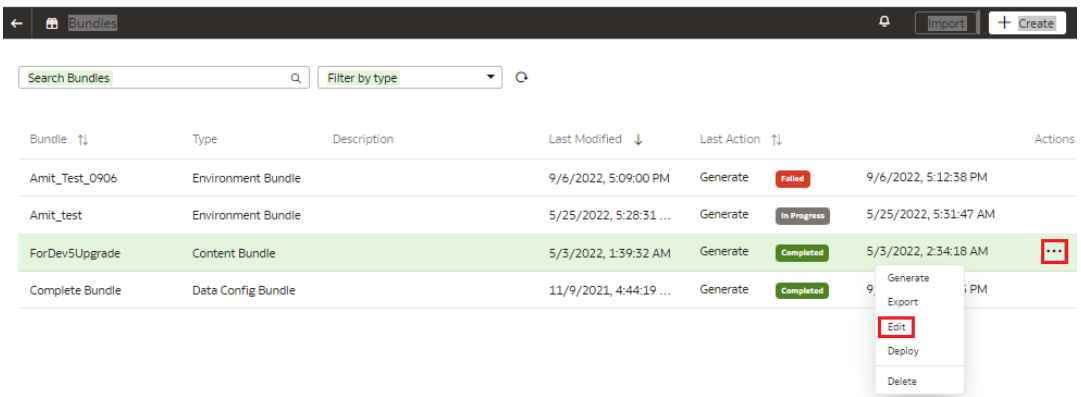


Edit a Bundle

Edit a bundle if you need to change the application artifacts captured in the bundle.

> **Note:**
>
> When you edit a bundle, you can't see the bundle definitions till you complete the bundle deployment. Functional areas and data augmentations in the bundle are visible after the bundle deployment process schedules them. The custom data configurations in the bundle are visible after the deploy custom data configurations process is completed.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Bundles** under **Application Administration**.
3. On the Bundles page, hover over the bundle that you created and click under **Actions**.
4. From **Actions**, select **Edit**.

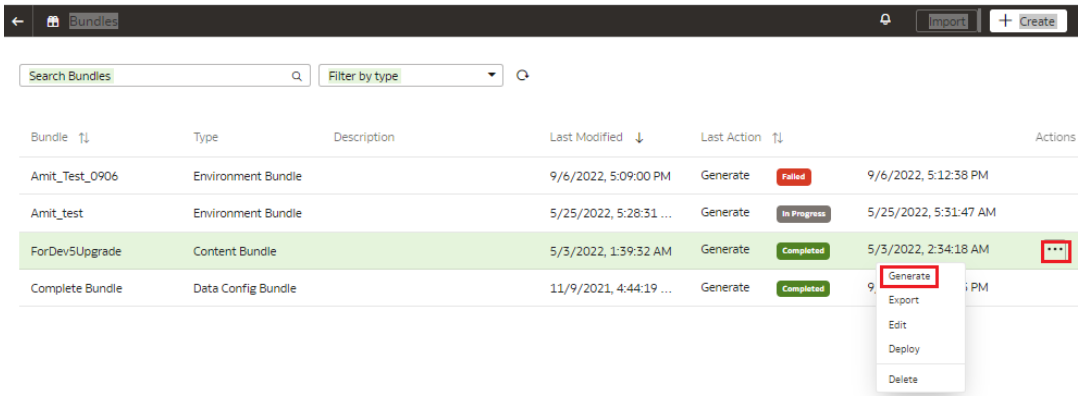


5. On the applicable bundles page, make your changes and then click **Update**.

Publish a Bundle

Publish a bundle from the source environment. This action generates a snapshot of the application artifacts and saves the snapshot to a repository. You can download this bundled artifact and import it into different instances.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Bundles** under **Application Administration**.
3. On the Bundles page, hover over the bundle that you created and click under **Actions**.
4. From **Actions**, select **Generate** to create and publish the bundle.



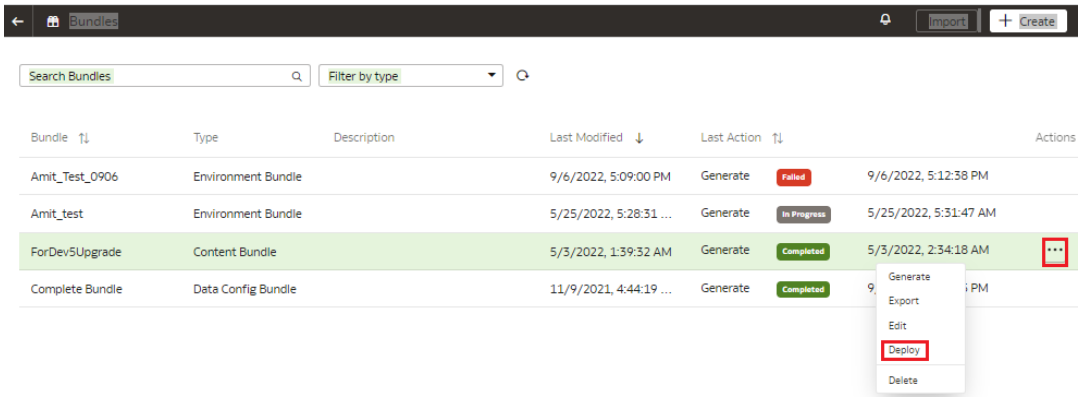
You see the bundle with Generated status on the Bundles page.

Deploy a Bundle

You can deploy a generated bundle in the target environment to revert to the state of artifacts represented by the bundle. The system validates the bundle before attempting any deployment to ensure software and model versions and any other dependencies are met.

For example, if you have a bundle A (created in instance A) and then later made some changes but want to revert those changes, then use the Deploy option. This option restores the artifacts to an older version by deploying the previous bundle in the system.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Bundles** under **Application Administration**.
3. On the Bundles page, hover over the bundle that you created and click under **Actions**.
4. From **Actions**, select **Deploy**.

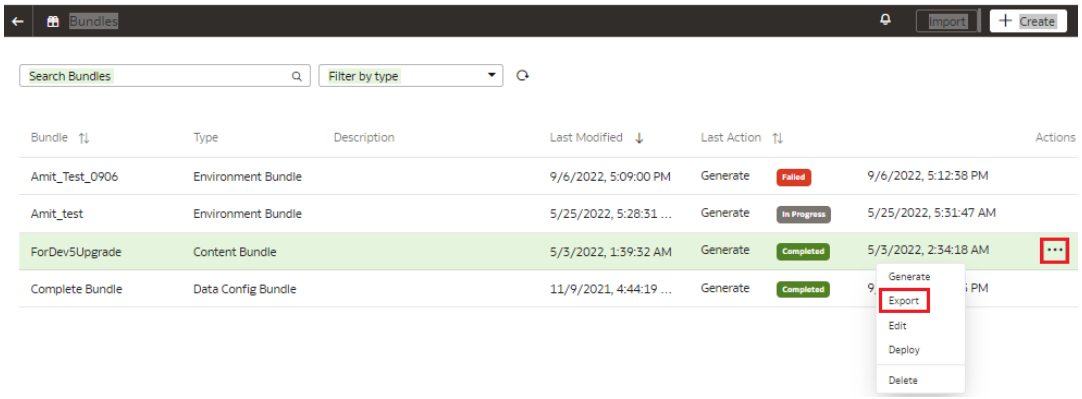


5. In the Deploy Bundle dialog, ensure that you see the bundle that you have selected.
6. Click **Deploy**.

Export a Bundle

Export the bundle .aab file from your source system to a repository or your local machine.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Bundles** under **Application Administration**.
3. On the Bundles page, hover over the bundle that you created and click under **Actions**.
4. From **Actions**, select **Export**.

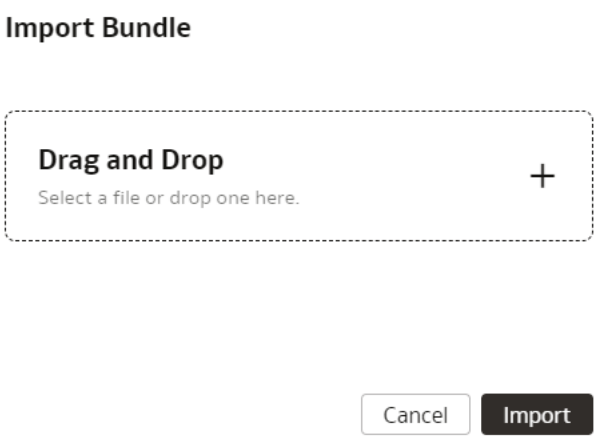


Import a Bundle

Import the bundle .aab file into the target environment from your computer to restore the state of the application to the checkpoint represented by the bundle.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Bundles** under **Application Administration**.
3. On the Bundles page, click **Import**.

4. In the Import Bundle dialog, click in Drag and Drop, and then select the applicable .aab file from your local machine.

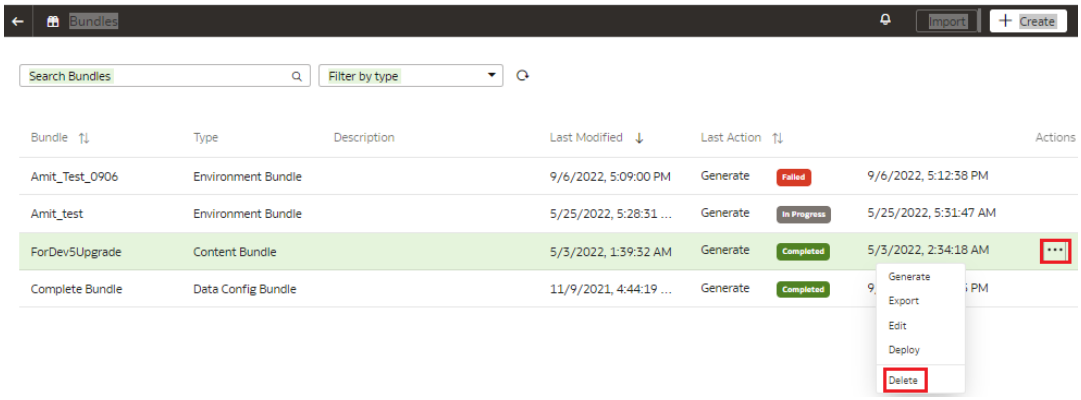


5. Click **Import**.

Delete a Bundle

Delete a bundle if you no longer require the snapshot of your application artifacts captured in the bundle.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Bundles** under **Application Administration**.
3. On the Bundles page, hover over the bundle that you created and click under **Actions**.
4. From **Actions**, select **Delete**.

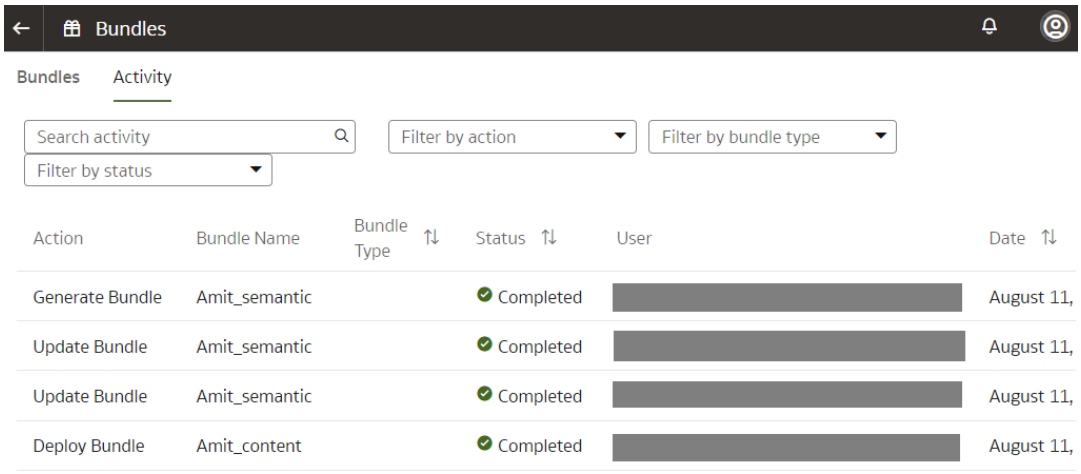


View the Activity History of Bundles

View all the bundles-related activities to understand the changes made to your instance, which bundles to use, and whether the existing bundles are still current. This information enables you to make informed decisions about creating updated bundles or deploying an existing bundle.

You see all the activities by bundle-related action, bundle names, bundle type, status, user who performed the activity, and date. You can organize the display by sorting columns, searching for activities, and filtering by action, bundle type, or status.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Bundles** under **Application Administration**.
3. On the Bundles page, click the **Activity** tab.



Configure a Virus Scanner

To keep Oracle NetSuite Analytics Warehouse virus-free, Oracle highly recommends that you set up the virus scanning servers used by your organization running on either 443 or 1708 ports only, to scan any files that are uploaded to Oracle NetSuite Analytics Warehouse.

When you configure virus scanning, the scanner checks all uploaded files including data files from the data pipeline and snapshots that you upload to restore content or to migrate content from another environment.

1. Sign in to your service.
2. In Oracle NetSuite Analytics Warehouse **Console**, click **Virus Scanner** under **Service Administration**.
3. On the Virus Scanner page, enter the host and port of the virus scanning server. For example, `my.virus.scanning.serverexample.com`.
4. Click **Save**.
5. To remove the current virus scanner configuration, click **Delete**.

6

Integrate Third-Party Data

Integrate data from other sources with your NetSuite data. Through this integration, you can create KPIs, metrics, cards, and dashboards using data from multiple sources.

Topics:

About Data Pipeline Options to Extract and Load

Use the information in this section to assess the data pipeline options to extract and load data.

Pipeline	Advantages	Limitations	License and Fees
Data Flow and Data Preparation	• Easy to use. Drag and drop data to create and schedule data flows. • No SQL code needed. • Select from a long list of connectors. • Intended for repetitive data requests. • Supports all standard databases.	• Can support only a few gigabytes of data. Limited to 1.1 million rows. • Incremental: Only based on one specific field.	Embedded in Oracle Analytics Cloud.
Oracle Cloud Infrastructure Data Integration	Supports large volumes of data.	Need to know how to browse data in Data Integration.	Additional license and fees required.
Existing ETL tool, if any	Ease of use, comfort and convenience.	General limitations of connectors.	Owned by customers.

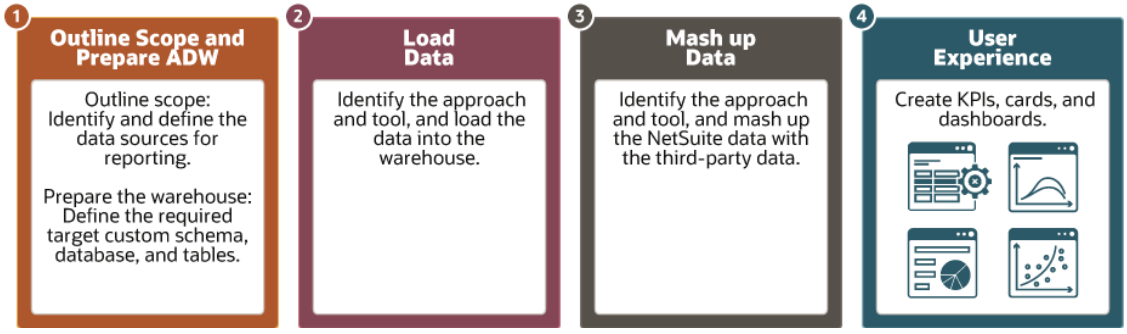
About Data Mashup and Transformation Options

Use the information in this section to assess the options to mash up and transform your data.

Mash Up the Data Layer	Advantages	Limitations
Extensibility Frameworks	• Robust solutions at semantic layer. • Best performance. Uses pre-calculated data.	• Need to be an expert user of the Oracle BI Server Administration tool. • Need to understand the existing metadata repository (RPD) file. • Must make minor changes to the metadata repository.
Data flow to collate NetSuite repository data and third-party data	• Can access the raw data in tables to create simple to complex calculated KPIs. • Provides ample scope for custom analytics. • Easy to extract and load data using drag and drop in data flow.	• Need to know about the tables in Oracle Autonomous Data Warehouse.
Data flow to collate NetSuite factory schema data and third-party data	Easy to extract and load data using drag and drop in data flow.	• Need to understand Oracle Autonomous Data Warehouse and the metadata repository. • Granularity limited to the data that's available.

Outline the Scope and Prepare the Warehouse for the Data

This section outlines the tasks you perform to prepare the autonomous data warehouse to receive the data.



You must ensure that the warehouse receives the right data in the right format by completing these tasks:

- Identify the data sources.
- Connect to the warehouse.
- Create the schema.
- Create the tables.

1. Identify and enlist the data source tables, fields, and rows to extract from the third-party source systems.
2. Examine the various approaches to connect to the warehouse to perform the data definition language (DDL) or data manipulation language (DML) operations and use the approach that suits your requirements.

3. After you connect to the warehouse, create a schema or database to absorb the data. If data is coming from multiple sources, decide whether you want to create multiple schemas or databases for different third-party sources or one schema or database for all third-party sources. See Provisioning Autonomous Data Warehouse Cloud
4. Create destination tables to store data from the incoming third-party sources.

   See Connecting SQL Developer and Creating Tables.

Load Data into the Warehouse

Perform these tasks to load data into Oracle Autonomous Data Warehouse.

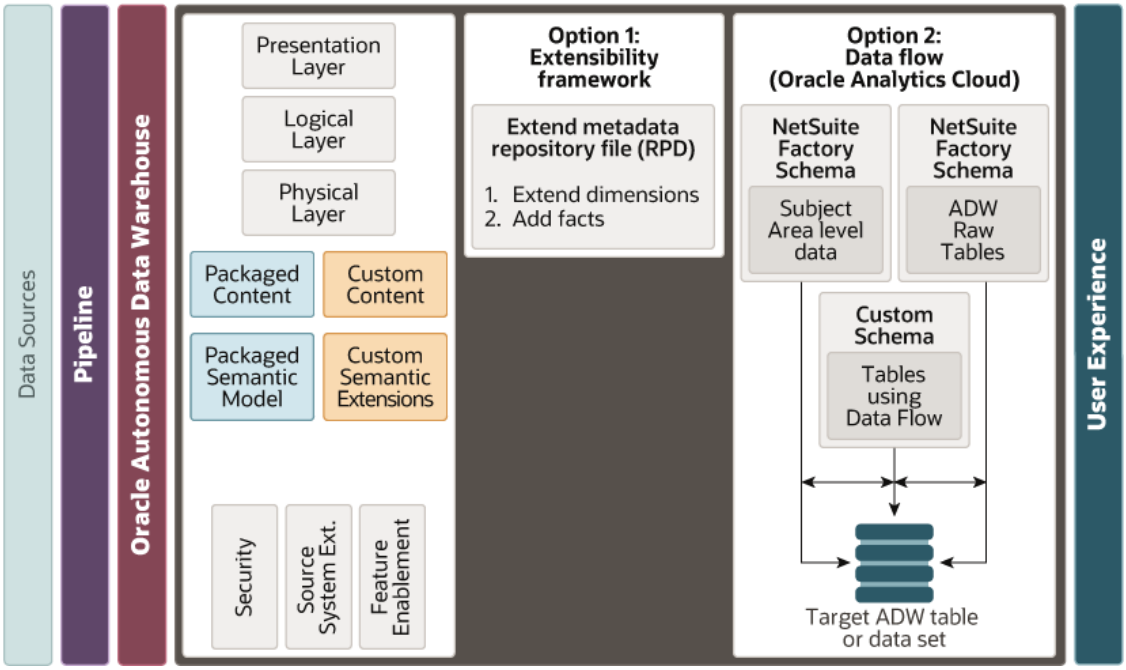
1. Consider these factors when selecting the pipeline tool for populating the warehouse with data:
   a. The data volume for the initial load.
   b. The data volume for incremental loads.
   c. The frequency of incremental loads.
   d. The complexity level of required transformations.
   e. The ETL tool in place, if any.
2. Based on the factors you considered while selecting the pipeline tool, evaluate these three options to select the tool:
   - Data Flow: This is a utility within the visualization area of Oracle NetSuite Analytics Warehouse that you can use to move data from various sources into the tables in the destination schema of Oracle Autonomous Data Warehouse. See Curate Your Data Using Data Flows.
   - Data Integration on Oracle Cloud Infrastructure: Data Integration is a fully managed, server-less, and native cloud service that helps you with common extract, load, and transform (ETL) tasks such as ingesting data from a variety of sources, cleansing, transforming, and reshaping your data, and efficiently loading it to target data sources on Oracle Cloud Infrastructure. See Data Integration.
   - Custom ETL: You can use any third-party ETL tool to move data into Oracle Autonomous Data Warehouse.
3. When the pipeline tool is ready, use it to load the data into Oracle Autonomous Data Warehouse.

Mash Up NetSuite Data with Third-Party Data

You might want to mash up data from multiple sources.

For instance, you might take web analytics data from warehouses of customer data, extract customer information from the Dynamix CRM, standardize and cleanse data, and then mash it up with NetSuite data. When data is in the autonomous data warehouse, you can select the

appropriate option for mashing up the data.



- Option 1: Extensibility Framework to Extend Models and Content
  You can use the Extensibility framework to extend models and content with data from other sources to your team without having to worry about upgrades. You must understand a star schema and know how to use or modify the metadata repository (RPD) file. See Customize Oracle NetSuite Analytics Warehouse and Extensibility Reference Architecture to get you started.
- Option 2: Data Flow to Mash Up Data in Oracle Autonomous Data Warehouse
  You can use this option to leverage third-party tables in Oracle Autonomous Data Warehouse (imported using one of the pipeline approaches) and NetSuite data loaded using an Oracle-managed pipeline. You can use a data flow to form a union, join, or filter, and transform data from the NetSuite schema and the custom schema that contains imported third-party data. See Curate Your Data Using Data Flows.

Create a Card or Dashboard

When you have the third-party data in Oracle Autonomous Data Warehouse and mashed-up with NetSuite data, you can create KPIs, KPI cards, decks, and dashboards, as described in the following topics:

- Create a KPI
- Create a Deck
- Create a Card
- Customize a Card
- Create Your First Analysis