Oracle
Primavera Analytics
Administration Guide for On-Premises

Version 23
July 2024

ORACLE®

Oracle Primavera Analytics Administration Guide for On-Premises

Copyright © 1999, 2024, Oracle and/or its affiliates.

Primary Author: Oracle Corporation

Contents

About this Guide

Once you have completed the tasks in the *Primavera Data Warehouse Installation and Configuration Guide* and the *Primavera Analytics Installation and Configuration Guide*, use this guide for advanced configurations and ongoing administration of Primavera Data Warehouse and Primavera Analytics. The guide also describes advanced techniques that require additional setup and configuration of both P6 EPPM and Primavera Unifier.

This guide assumes that the following are true (as applicable to your deployment):

▶ The current version of Primavera Data Warehouse is installed and configured and the STARETL process is running successfully.
▶ The current version of Primavera Analytics is installed and configured, and access by Oracle Business Intelligence (OBI) is successful.
▶ The current version of P6 EPPM is installed and running successfully.
▶ The current version of Primavera Unifier is installed and running successfully.

For On-Premises

Use either the PL/SQL-based or ODI-based StarETL process.

> **Note**: The term, StarETL, when used generically, applies to PL/SQL as well as ODI.

In This Section

About Primavera Analytics

Oracle's Primavera Analytics is a packaged business intelligence solution that provides valuable insights into your projects and portfolios in Oracle's Primavera P6 Enterprise Project Portfolio Management (P6 EPPM), Primavera Unifier, and Oracle Primavera Cloud. Primavera Analytics is built upon the industry leading Oracle Business Intelligence Suite (OBI) application.

Primavera Analytics consists of a:

▶ Repository Definition

The Primavera Analytics Repository Definition contains all of the information OBI needs to allow end users to create purposeful analysis based on data contained in the data warehouse. It also contains information such as how OBI connects to the Primavera Data Warehouse, how security is applied to each user, and how to join fact and dimension tables.

▶ Sample catalog

The Primavera Analytics sample catalog contains over 450 pre-built analyses and dashboards. They are organized in a catalog structure to help facilitate the capabilities available by combining P6 EPPM, Primavera Unifier, and/or Oracle Primavera Cloud with OBI.

About Primavera Data Warehouse

Primavera Data Warehouse is a data repository that collects and stores data from the following products:

- P6 EPPM
- Primavera Unifier
- Oracle Primavera Cloud

It contains a process called STARETL to *extract*, *transform*, and *load* (ETL) data from the above applications into Primavera Data Warehouse.

Primavera Data Warehouse separates the data into:

- Facts

  Fact tables contain business facts or measures. For example: Actual Total Cost, Workflow Duration.

- Dimensions

  Dimension tables contain descriptive attributes or fields. For example: Activity Name, Cost Breakdown Structure.

As Primavera Data Warehouse schema accumulates project and cash flow data, it provides baselines for tracking trends and for advanced business intelligence. It also provides key metrics on current data, which allow you to drill through root cause analysis.

About the Analytics Administration Application

The Analytics Administration application gives administrators the ability to add or change settings for their deployment of Primavera Data Warehouse. Use this administration application to perform the following tasks:

- Run on-demand or schedule STARETL runs
- Configure multiple data sources, codes, and UDFs for P6 and Primavera Cloud
- View the history and logs of previous STARETL runs

The Analytics Administration application enables you to administer Analytics from any location.

Product Accessibility

This topic includes guidelines and tips regarding accessible product usage.

Oracle has also published an Accessibility Conformance Report (ACR) for the Primavera Analytics product. The latest version of the Primavera Analytics Accessibility Conformance Report can be accessed from Oracle's Accessibility Program Website at this link: ***https://www.oracle.com/corporate/accessibility/vpats.html#acrs-tab*** (***https://www.oracle.com/corporate/accessibility/vpats.html#acrs-tab***).

Assistive Technology

Users can use assistive technology, such as screen readers, with Primavera Analytics.

Keyboard Shortcuts

The list of keyboard shortcuts for Primavera Analytics, including page-specific and browser-specific keyboard shortcuts, can be accessed here: ***Keyboard Shortcuts*** (on page 7)

Keyboard Shortcuts

KeYou can use your keyboard to navigate throughout Primavera Analytics and the Analytics Administration application, and access frequently used or important functionality. Use the **Tab** key to cycle through elements within the application, and press Enter or Spacebar to access them. Some keyboard shortcuts may change depending on the browser you use.

Common Shortcuts

The following shortcuts are common to all pages of Primavera Analytics:

▸ Press **Tab** to navigate to any element.
▸ Press **Enter** to click on the element.
▸ Press the space bar to select a check box.
▸ Press **Shift+Tab** to navigate back to any element.
▸ Use the Arrow keys(LEFT and RIGHT) to select radio buttons/within the table row.
▸ Use Arrow keys(UP and DOWN) to move through the drop-down lists/ to select values in combo boxes.

Managing Personal Information

This chapter describes how to manage personal information (PI) in Primavera Analytics.

In This Section

About Consent Notices

Consent notices inform users how personal information (PI) is collected, processed, stored, and transmitted, along with details related to applicable regulations and policies. Consent notices also alert users that the action they are taking may risk exposing PI. Primavera products helps you to ensure that you have requested the appropriate consent to collect, process, store, and transmit the PI your organization holds as part of any source application (P6 and Unifier) that can be configured with Primavera Analytics.

About Personal Information

Personal information (PI) is any piece of data which can be used on its own or with other information to identify, contact or locate an individual or identify an individual in context. This information is not limited to a person's name, address, and contact details. For example, a person's IP address, phone IMEI number, gender, and location at a particular time could all be personal information. Depending on local data protection laws , organizations may be responsible for ensuring the privacy of PI wherever it is stored, including in back-ups, locally stored downloads, and data stored in development environments.

As part of your Primavera Analytics Cloud Service, you may be using an identity management domain to manage your user access and entitlements across a number of cloud and on-premises applications and services. If you are using or accessing an identity management domain you are responsible for deleting your details and data from the identity management domain. You are responsible for retrieving your content in the identity management domain during your applicable services period.

Configuring Consent Notices for Analytics

Analytics consumes data from source applications and makes it available to Analytics users. End-users must give their consent in the *source* application to read and agree to the consent message to ensure they

▸ have access to the data in Analytics.

▶ understand the responsibilities with regard to data protection and security.

The following source applications can be configured with Analytics:

▶ P6 EPPM

For detailed instructions on how to configure consent notices in P6 EPPM, refer to *P6 EPPM Application Administration Guide.*

▶ Primavera Unifier

For detailed instructions on how to configure consent notices in Primavera Unifier, refer to *Primavera Unifier Administration Guide*

▶ Oracle Primavera Cloud

Your Responsibilities

Information security and privacy laws can carry heavy penalties and fines for organizations which do not adequately protect PI they gather and store. Data visible to an analytics user depends on the consent notices configured and accepted by users in the source applications which can be configured with Analytics.

If these laws apply to your organization, it is your responsibility to ensure consent notices are configured in the source applications before they are required.

Ensuring Privacy of Data Collection

Personal information (PI) in Primavera Analytics depends on the changes made in source applications such as P6 EPPM, Primavera Unifier . When PI data is modified or deleted in the source applications, run the ETL process to ensure it is automatically reflected in Analytics. If a user is deleted in a source application, they can no longer access Primavera Analytics.

Ensure the user is also deleted from the historical data of Primavera Analytics.

Limiting Granular Access to Data

Products provide granular access controls, by record, data element, type of data, or logs. Ensure Primavera Data Warehouse adheres to the same user privileges and access rights as P6 EPPM and Primavera Unifier.

Ensuring Data Purging and Data Deletion

Primavera Analytics is a Data Warehouse application. Data is not purged because Primavera Analytics maintains data history. A **Delete** option is available if needed but not available to all users.

ODI logging is purged weekly. Use the **data_cleanup_package** to purge the data for specific data source and resources on demand.

Ensuring Data Masking is Supported

Data masking is the ability to display only a portion of a data field or prevent viewing data. Oracle database security policies can hide access to certain columns. Ensure you review this feature for Primavera Analytics.

Ensuring IP Allowlists

An IP allowlist is a list of trusted IP addresses from which your users can access domains. Primavera Analytics uses OBIEE for creating analyses and dashboards. For more details on IP allowlists, refer to OBIEE documentation.

Controlling Access to Special Categories of Data

Primavera Analytics has control over some sensitive fields. Control can be regulated to only users that are allowed to login. Data access is generally project level and either cost or no cost, and granularity is provided for those differences. Global data is also visible but honors same security as source applications.

Administering Analytics

In This Section

Configuring P6 EPPM for Analytics

This chapter describes how to configure P6 EPPM settings for Primavera Analytics.

In This Section

Configuring P6 EPPM History Settings for Primavera Data Warehouse

To capture and store historical data from P6 EPPM in Primavera Data Warehouse, configure the history settings as follows:

1) Configure P6 EPPM to specify the History Level and History Interval for each project:
    a. Sign in into P6 with a user that has the Edit Project Details Except Cost/Financials privilege.
    b. Click **Projects**.
    c. On the **Projects** navigation bar, click **EPS**.
    d. On the **EPS** page:
        1. Select a project.
        2. From the **Actions** ▼ menu and select 🗔 **Set Project Preferences**.
    e. In the **Project Preferences** pane, click **Analytics**.
    f. On the **Analytics** page:
        1. Select one of the following **History Level** settings:
            ▪ **None** (default)
            ▪ **Project**
            ▪ **WBS**
            ▪ **Activity (Daily)**
            ▪ **Activity (Weekly)**
            ▪ **Activity (Monthly)**
        1. Select one of the following **History Interval** settings:
            ▪ **Month** (default)
            ▪ **Week**
            ▪ **Quarter**
            ▪ **Year**
            ▪ **Financial Period**

1. Select one of the following **Data Warehouse Update Frequency** settings:
   - **Scheduled**
   - **Immediate**
g. Click **Save** or **Save and Close**.

> **Note**: If you select **Activity (Daily)** as the **History Level**, the **History Interval** for Activity and Resource Assignment history is automatically set to **Daily**. The History Interval will also match **Activity (Weekly)** and **Activity (Monthly)** if either are selected, setting Resource Assignment history to **Weekly** and **Monthly**, respectively. Use caution when selecting which projects you choose to have daily history captures as this can affect the amount of time and the volume of data processed when running the STARETL process. You still need to set the history interval level when selecting Activity level history, which will apply to project and WBS-level history only.

2) Publish the project data using the publication services in both P6 EPPM and Primavera Unifier.
3) Run the STARETL process at the lowest time interval for which history is being captured:
   a. Go to **<PDW Install Folder>\star\etl_homes\staretl<id>**.
   b. Run **staretl.bat** (with Windows) or **staretl.sh** (with UNIX or Linux).

Configuring Oracle Data Integrator (ODI) Settings for P6 EPPM Data

Primavera Data Warehouse accumulates data over time with each run of the STARETL process. The traditional ETL process is used to run STARETLs to extract, transform, and load (ETL) data into Primavera Data Warehouse.

Primavera Analytics gives you an alternative option to run STARETLs by using Oracle Data Integrator (ODI) to extract, transform, and load (ETL) data from the following data sources into Primavera Data Warehouse:

▶ P6 EPPM
▶ Unifier

Oracle Data Integrator (ODI) is a comprehensive data integration platform that covers all data integration requirements: from high-volume, high-performance batch loads to event-driven, trickle-feed integration processes and SOA-enabled data services. ODI allows for near real-time migration of data.

ODI allows for features not supported by the traditional ETL process such as:

▶ Real-time analytics
▶ Single project loading on-demand

> **Note**: ODI is a separately licensed application.

In This Section

Enabling P6 EPPM Projects for Real-Time Analytics

Oracle Data Integrator (ODI) provides support for real-time analytics, which enables project changes to be immediately sent over to the Primavera Data Warehouse instead of waiting for the next scheduled ETL. Once a project is selected for real-time analytics, it is no longer part of the scheduled ETL.

> **Note**: Primavera Unifier projects cannot be enabled for real-time analytics.

To enable a P6 EPPM project for real-time analytics, complete the following steps:

1) Sign in to P6 EPPM.
2) Click the **Projects** tab.
3) Click the **EPS** page.
4) Locate the project you want to enable for real-time analytics.
5) Click the gear icon ⚙ to expand the menu, and select **Set Project Preferences**.
6) In the **Project Preferences** window, click the **Analytics & Services** tab.

7) In the **Analytics** section, expand the **Data Warehouse Update Frequency (ODI Only)** drop-down menu and change the setting to **Immediate**.
8) Click **OK** to save the project preferences.

Repeat these steps as needed to enable for real-time analytics for other P6 EPPM projects.

Using Slowly Changing Dimensions and Daily Snapshot Fact Tables for P6 EPPM Data

To capture history data in P6 projects and enable predictions in CIC, it is necessary to enable Slowly changing dimensions (also known as Type 2 dimensions) to preserve an attribute's history of values by capturing non-fact or dimensional data and associating that data with the correct version of the fact row. Captured data includes project, resource, activity, resource assignment dimensional data, and more. Slowly changing dimensions are used when is it important to preserve the historic context of facts.

When source data elements change or when the STARETL process is run, a new row is added to the dimension table and the previous row is left unchanged. Previously recorded facts remain associated with the old dimension row and new facts are associated with the new dimension row. These rows show the changes to an attribute over time. For example, if the Planned Start Date for an activity has changed between the current capture and the previous capture, the old and new date would both be recorded in the dimension table.

Dimension rows in the dimension table are distinguished by the **Effective Start Date**, **Effective End Date**, and **Current Flag**. The date and time the STARETL process starts becomes the **Effective Start Date** for the new dimension row. The new version of a dimension **Effective End Date** defaults to 01-Jan-3000 00:00:00. When the STARETL process is run again, the **Effective End Date** for the latest version of the attribute then becomes the **Effective Start Date** for the current version minus one second.

For example, if the **Effective Start Date** for the current version of the attribute is 04-Nov-2019 9:12:28 PM, then the **Effective End Date** for the previous version becomes 04-Nov-2019 9:12:27 PM. The most recent dimension row in the dimension table has a value of "1" in the **Current Flag** column. All other rows have "0" in the **Current Flag** column.

Dimension Table

Project Id	Project Name	At Completion Units	Effective Start Date	Effective End Date	Current Flag
IT00999	Project Silicon	6164.00	10/31/2019 9:22:53 PM	11/1/2019 9:11:39 PM	0.00
		6176.00	11/1/2019 9:11:40 PM	11/2/2019 9:21:56 PM	0.00
		6180.00	11/2/2019 9:21:57 PM	11/3/2019 9:07:33 PM	0.00
		6172.00	11/3/2019 9:07:34 PM	11/4/2019 9:12:27 PM	0.00
		6091.00	11/4/2019 9:12:28 PM	1/1/3000 12:00:00 AM	1.00

Daily snapshot fact tables enable the daily capture of metrics at the Resource Assignment and Activity Levels. Daily snapshot fact tables are used in conjunction with slowly changing dimensions. For example, when the Activity History daily snapshot is recorded, new fact rows will be updated with the most current version of each record in the activity dimension. Previously recorded facts remain associated with the old dimension row version.

The daily snapshot makes it easy to determine what metrics such as Remaining Units looked like at any point in time across all projects that have been opted into this level of data capture.

Enabling Daily History Snapshots and Slowly Changing Dimensions

To enable SCDs and daily history snapshots for a P6 project:

1) Sign in to P6 EPPM.

   http://<host>:<port>/p6/action/login

2) In P6, click **Projects**.

3) On the **Projects** screen navigation bar, click **EPS**.

4) On the **EPS** page:

   a. Select a project.

   b. From the ⚙ **Row Actions** menu of the selected project, select **Set Project Preferences**.

5) In the **Project Preferences** pane:

   a. Ensure the project is configured for publication services.

      1. Click the **Analytics & Services** tab.

      2. Select the **Enable Publication** option in the **Publication** section.

   b. Configure the project's History Level setting.

      1. Click the **Analytics & Services** tab.

      2. From the **History Level** list, select *Activity (Daily), Activity (Weekly)* or *Activity (Monthly).*

         Note: You should still configure the **History Interval** setting, but it will only apply to Project and WBS-level history metrics.

   c. Click **Save and Close**.

Turning Off Slowly Changing Dimensions

If you turn off the slowly changing dimensions for a field, the current dimensional row can be updated with a new value without the addition of a new dimensional row.

You can control slowly changing dimensions for a data source on a field-by-field level by making adjustments to **mappings.tcsv** for that data source, which is in **<PDW install Folder>\star\etl_homes\staretl<id>\res**.

A Y or N flag for each field in the mappings file controls whether a change to this field would cause a new row to be added to the dimensional table. Y indicates that a change will cause a new row to be added. N indicates that a new row will not be added if a change is made to this field. For example, if the Activity Description field has a Y flag, then changes made to this field will cause a new row to be added to the dimensional table the next time the STARETL process is run. If the EPS Name field has an N flag, then the STARETL process will not update the dimensional table with a new row. Configuring the Y or N flag for each field can prevent cascading updates to fact tables. For example, if a change is made to the EPS table, that change would cause a new row to be added to the dimension table for all of the activities affected by the change.

Using Burn Down for P6 EPPM Data

Primavera Data Warehouse and Primavera Analytics contain an additional way that data can be analyzed outside of the standard P6 EPPM methods. Primavera Data Warehouse and Primavera Analytics include additional tables and configuration for Burn Down. In the simplest of terms, Burn Down is a set of calculations that display how a project will progress if everything goes "according to plan". Primavera Analytics includes metrics and Key Performance Indicators (KPIs) for both the current project data, as well as any point in the past for which the data was captured. These metrics and KPIs include both current and historical activity counts, units, and costs. Primavera Analytics and Primavera Data Warehouse has the ability to track and monitor how the current schedule is executing in relationship to how it was supposed to progress at the start of the project (T-0). This includes, but is not limited to, any new activities that were added to the current project that did not exist at the start of the project (emergent), activities that have been removed in the current schedule that existed in the point in time capture of the project (deleted), and activities whose start or finish has changed since the point in time capture of the project.

Use the sections in this chapter to learn more about the additional functionality, how to set it up in P6 EPPM, and how to work with the data in Primavera Analytics.

> **Note**: Oracle provides several white papers that detail aspects of Burn Down:
>
> - Burn Down Detail Data Flow:
>   http://www.oracle.com/webfolder/technetwork/tutorials/primavera/OnlineLearning/WhitePapers/BurnDown_Details_DataFlow.pdf
> - Burn Down Units Details:
>   http://www.oracle.com/webfolder/technetwork/tutorials/primavera/OnlineLearning/WhitePapers/BurnDown_Details_Units.pdf
> - Burn Down Counts Details:
>   http://www.oracle.com/webfolder/technetwork/tutorials/primavera/OnlineLearning/WhitePapers/BurnDown_Details_Counts.pdf

Creating Project UDFs for Burn Down Subject Areas

Only P6 EPPM projects can be manually included to take advantage of the Oracle Primavera Burn Down tables and functionality. This section describes how to create and configure the necessary P6 EPPM project level user-defined fields (UDFs) that the STARETL process needs to include project(s) in Burn Down.

To create the project UDFs, complete the following steps:

1) Sign in to P6 EPPM with a user account that has the privilege to change enterprise data (for example, admin).
2) From the **Administer** ▼ menu and select **Enterprise Data**.
3) On the **Enterprise Data** page, expand **Projects** and select **Project UDFs**.
4) On the Project UDFs page:
   a. Add a **sys_workdown** UDF with a data type of **Text**.
   b. Add a **sys_workdown_date** UDF with a data type of **Start Date**.

Selecting the Project for the Burn Down Subject Area

For a project to be included in the Burn Down tables and subject area, you must configure specific project UDF and History Level settings in P6.

To select which P6 EPPM project will be included in Burn Down tables and metrics, complete the following steps:

1) Sign in to P6 EPPM.
2) In P6, click **Projects**.
3) On the **Projects** navigation bar, click **EPS**.
4) On the **EPS** page:
   a. Select a project.
   b. From the **Actions** ▼ menu, select 🔲 **Set Project Preferences**.
5) In the **Project Preferences** pane, click **Analytics**.
6) On the **Analytics** page, click the **History Level** list, and then select **Activity**.
7) In the **Project Preferences** dialog box, click **Save and Close**.
8) On the **Projects** navigation bar, click **EPS**.
9) On the **EPS** page, add the **sys_workdown** and **sys_workdown_date** UDF columns to the page.
10) For the projects you want to add to the Burn Down subject area:
   a. Enter `project` in the **sys_workdown** column.
   b. Enter a date value (for example, 09/30/2015) in the **sys_workdown_date** column. This value is used as the snapshot date for comparison to actual project values. If no value is entered here, the **Project Start Date** is used.

> **Note**: The date used for the initial Burn Down data capture is one day before the date entered for this UDF. For example, if the

sys_workdown_date is 03/31/2016, then the STARETL process uses 03/30/2016 for the Burn Down comparison.

Burn Down and Work Planning with the STARETL Process

After adding P6 EPPM projects for inclusion in the Burn Down and/or Work Planning subject areas, it is important that the date/time period specified in P6 EPPM under **Administer**, **Application Settings**, **Services** in the **Publication Period** section is later than the latest project finish for ALL project(s) that will be included for Burn Down and/or Work Planning. For example, in P6 EPPM Publication Services settings, if you specify a start date of October 1, 2015 and the finish date is current date plus setting is two years, the finish date for all of the Burn Down/Work Planning project(s) must have a finish date before September 30, 2017. If a project extends past the finish date setting in the publication services settings, the Burn Down/Work Planning portion of the STARETL process generates an error.

> **Note**: Setting a finish date range that is too long will affect the STARETL process run time. Each day the publication services are run, the finish date range also extends by one day.

Scheduling a Burn Down Project

A project can have activities scheduled and completed before a burn down begins. This gives users the flexibility to begin the project burn down on the date specified in the **sys_workdown_date** UDF rather than the **Project Start Date**.

In a Burn Down project, the initial snapshot of the project's activities and assignments is taken the day before the date entered in the **sys_workdown_date** UDF. This snapshot is used as a baseline to compare against actual values. For example, the **Baseline Remaining Units Burned** can be compared to the **Remaining Units Burned**. The STARETL process must be run one day before the **sys_workdown_date** (or the **Project Start Date** if this UDF is not used) to capture this initial snapshot.
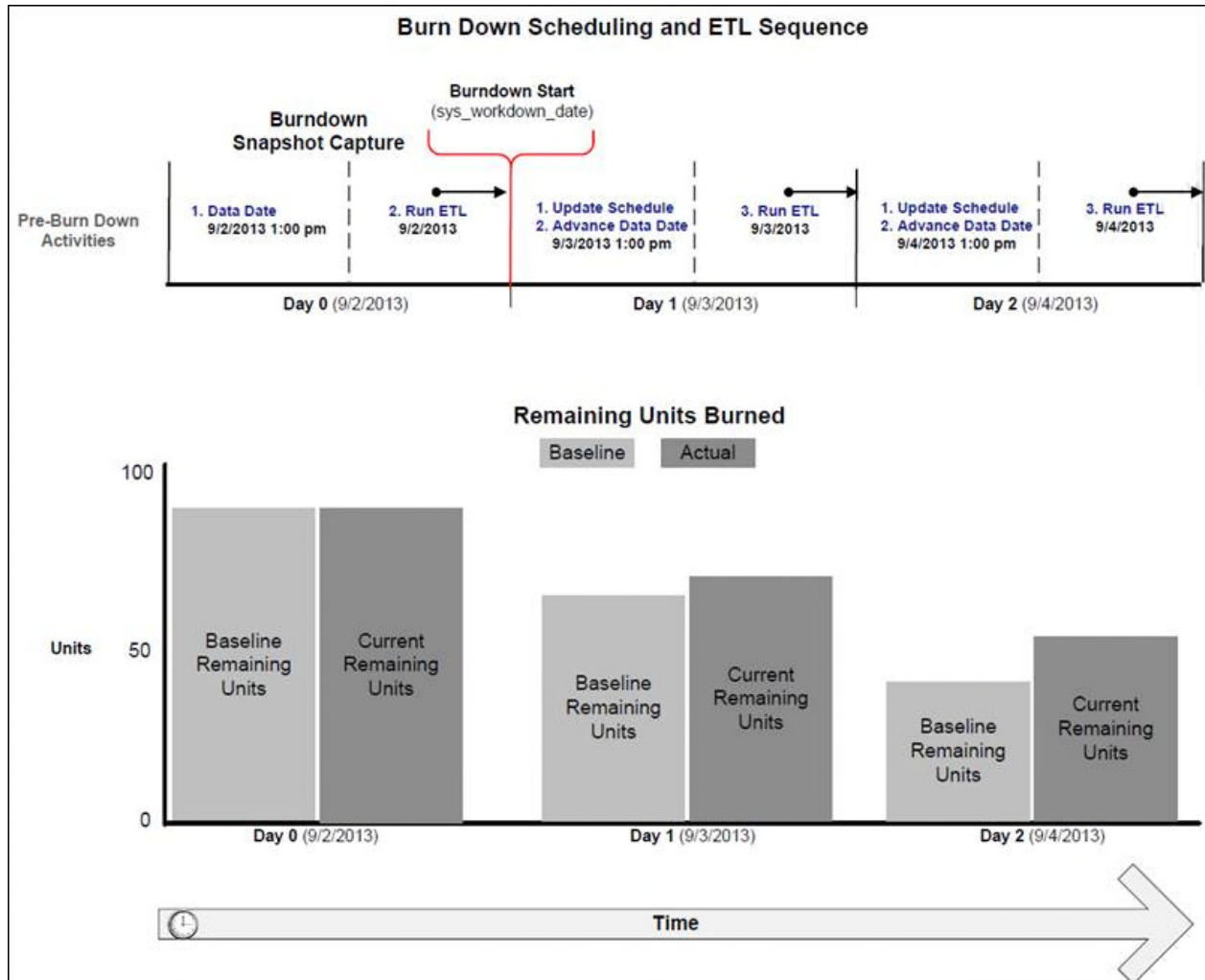
About the STARETL Process and Burn Down Metrics

When using Burn Down Metrics, you must update the schedule and run the STARETL process. Each daily STARETL process updates two sets of daily metrics:

▸ **Metrics for 'Today'**: All work completed today through the latest schedule update and data date change is recorded as work completed for the current date when the STARETL process occurs.

▸ **Metrics for 'Yesterday'**: Since the data from yesterday is only current through the latest schedule update and data date change, any additional work completed yesterday but AFTER the update is now added to the daily metrics for 'yesterday'.

For example, a project is updated at 6pm daily and then the STARETL process is run. The next day, these become metrics for 'yesterday' and initially reflect work only through 6pm yesterday. When the STARETL process is run today, any additional work completed between 6pm and midnight is now included under the metrics for 'yesterday'.

Calculating Metrics in the Burn Down Subject Area

To capture daily trending information, complete the following steps each day to calculate the metrics in the Burn Down subject area:

1) Update the status of activities in the projects you configured in P6 to be included in the Burn Down subject area. For example, update the Actual and Remaining Units and which activities are completed.
2) Adjust the data date to correspond to the latest status update.
3) Run the STARETL process.

> **Note**: The STARETL run must occur within each calendar date to properly capture daily metrics.

Using Work Planning for P6 EPPM Data

Primavera Analytics supports Work Planning for P6 EPPM data. Work Planning is a method of tracking how work is progressing based on a set of user-determined points in time. Similar to Burn Down, Work Planning also uses project-level user-defined fields (UDFs) to select the P6 EPPM projects included in the Work Planning calculations and metrics and also the point in time you want to use for the comparison.

Creating Project UDFs for Work Planning Subject Areas

You can manually include P6 EPPM projects to take advantage of the Work Planning tables and functionality.

You must create and configure the necessary P6 EPPM project-level user-defined fields (UDFs) that the STARETL process needs to include projects in Work Planning. To do this, complete the following steps:

1) Sign in to P6 EPPM.
2) From the **Administration** ▾ menu, select **Enterprise Data**.
3) On the **Enterprise Data** page, expand **Projects** and select **Project UDFs**.
4) On the **Project UDFs** page:

a. Add a **sys_workprocess** UDF with a data type of **Text**.

b. Add a **sys_scope_freeze** UDF with a data type of **Text**.

c. Add a **sys_schedule_freeze** UDF with a data type of **Text**.

Selecting a Project for the Work Planning Subject Area

The Work Planning subject area is designed for ongoing project work typical of routine and online maintenance projects. In these projects, planning metrics are tracked week over week and workweek metrics are tracked daily. There are specific Project UDF and History Level settings required in P6 to include a project in the Work Planning subject area.

Complete the following steps to select the P6 EPPM projects for the work planning subject area:

1) Sign in to P6 EPPM.
2) In P6, click **Projects**.
3) On the **Projects** navigation bar, click **EPS**.
4) On the **EPS** page:
   a. Select a project.
   b. From the **Actions** ▼ menu, select 🗔 **Set Project Preferences**.
5) In the **Project Preferences** dialog box:
   a. Ensure the project is configured for publication services.
      1. In the **Project Preferences** pane, click **Services**.
      2. On the **Services** page, in the **Publication** pane, select the **Enable Publication** option.
   b. Configure the project's History Level setting.
      1. In the **Project Preferences** pane, click **Analytics**.
      2. On the **Analytics** page, in the **History Level** list, select **Activity**.
   c. Click **Save and Close**.
6) On the **Projects** navigation bar, click **EPS**.
7) On the **EPS** page, add the **sys_workprocess**, **sys_scope_freeze**, and **sys_schedule_freeze** UDF columns to the page.
8) For the projects you want to add to the Work Planning subject area:
   a. Enter `workweek` in the **sys_workprocess** column.

> **Note**: Use the format `T-<x>w` (where `<x>` equals the number of weeks prior to scope freeze and schedule freeze) must be followed exactly when entering the **sys_scope_freeze** and **sys_schedule_freeze** UDF values.

b. Enter a value in the **sys_scope_freeze** column. This is a relative value for each week in a project's schedule. For example, if you enter `T-2w`, scope freeze for each week in a project is calculated as two weeks prior to the Planned Start Date of that week. Scope freeze metrics, such as Scope Freeze New or Scope Freeze Lost, are captured for each week in a project schedule from each week's calculated scope freeze date up to and including its completion.

c. Enter a value in the **sys_schedule_freeze** column. This is a relative value for each week in a project's schedule. For example, if you enter `T-1w`, schedule freeze for each week in a project is calculated as one week prior to the Planned Start Date of that week. Schedule freeze metrics, such as Schedule Freeze New or Schedule Freeze Lost, are captured for each week in a project schedule from each week's calculated schedule freeze date up to and including its completion.

Scheduling a Work Planning Project

Scope freeze, schedule freeze, and actual metrics are captured for each week of planned work in a project that has been opted into the Work Planning subject area. Each individual week in the schedule will have these metrics captured leading up to that week's execution, allowing users to monitor their planning performance on a weekly basis for an upcoming week of work.

The **sys_scope_freeze** and **sys_schedule_freeze** UDFs control the amount of history or the number of weeks these metrics capture for each individual week leading up to the execution week.

The diagram below illustrates how the ETL process would capture Work Planning metrics for the week 12/17/2012 to 12/23/2012 given the following P6 settings:

▸ The **sys_scope_freeze** equals **T-2w**
▸ The **sys_schedule_freeze** equals **T-1w**
▸ The **Starting Day of Week** equals **Monday** (this is set in the P6 Application Settings)

In this scenario, the **Scope Freeze** for the week starting on Monday 12/17/2012 occurs two weeks prior (T-2w) to that week's planned start date. **Schedule Freeze** occurs one week prior (T-1w) to that week's planned start date. Daily Actual Start and Finish metrics are captured throughout the week.

Once the STARETL process has run on these respective freeze dates, any changes in the schedule for an upcoming week (such as adds, deletes, or planned date changes) are recorded the next time the STARETL process runs.
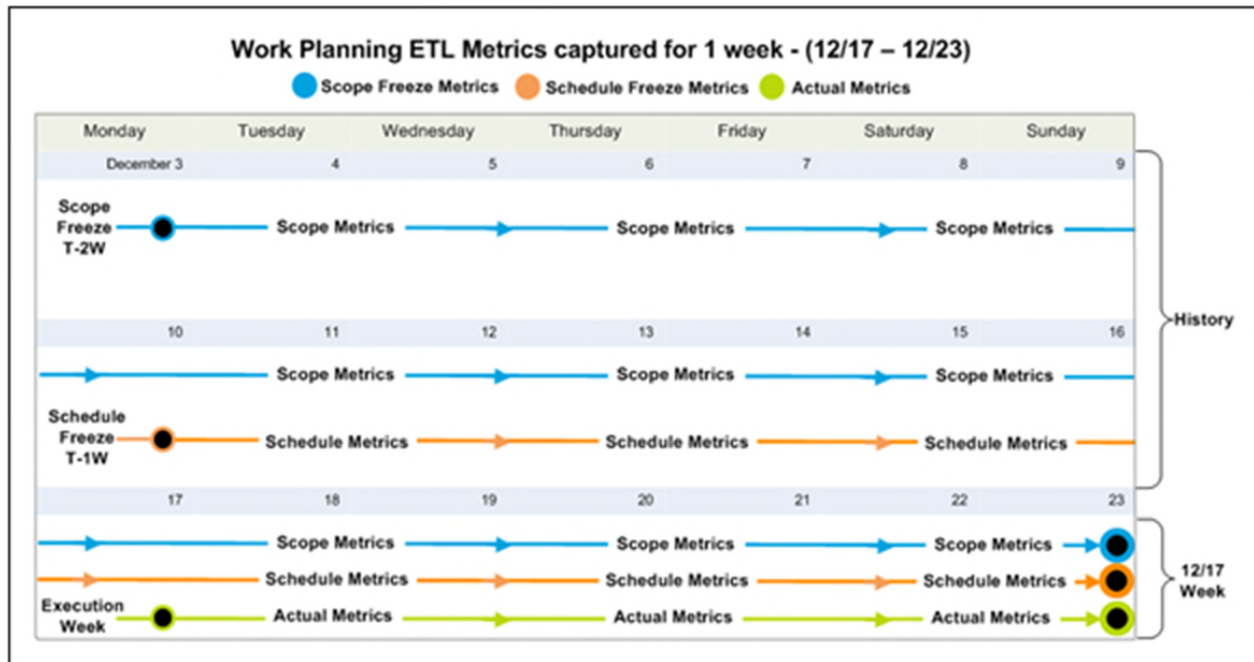
Using Graded Approach for Schedule Adherence

Graded approach to schedule adherence is a method for tracking adherence using four distinct levels of work scheduling (A, B, C, and D).

The levels of work scheduling are:

▶ **A (Hourly)**: Used when you require the most control, adherence, and management oversight. Reasons include risk (probability safety analysis/risk assessment), personnel safety, or other significant concerns. Users must finish Level A activities within one hour of the target planned finish date.

▶ **B (Daily)**: Used for work requiring multi-discipline coordination, significant tagouts, or risk significant work. Users must finish Level B activities on the same day as the target planned finish date.

▶ **C (Weekly)**: Used for non-risk significant and routine work that does not require support from other work groups. Users must finish Level C activities within the week of the target planned finish date.

▶ **D (No Tracking)**: Used for housekeeping, shop work, and other types of work that have no impact to the plant. Level D activities are considered "fill-in" work and are not tracked for schedule adherence.

For graded schedule adherence, ensure history exists for the project as of the scheduled freeze date and T-0 baseline date for the week the STARETL process will run.

During T-0 for a given week when you run the STARETL process, the Baseline Planned Finish Date (as of T-0) and the Schedule Freeze Planned Finish Date (as of schedule freeze UDF) for the activities will be captured. You can use these dates along with the assigned adherence grade to calculate the graded schedule adherence in CIC Analytics.

Configuring Primavera Unifier for Primavera Data Warehouse

Primavera Data Warehouse relies on data from Primavera Unifier. Before using Primavera Data Warehouse, configure and run Primavera Unifier Publication Services to ensure Unifier current data is ready to be extracted, transformed, and loaded (ETL) in Primavera Data Warehouse.

This chapter describes how to set up Primavera Unifier for Primavera Data Warehouse.

In This Section

Configuring and Running Primavera Unifier Publication Services

If you are using Primavera Unifier as a data source, Primavera Data Warehouse relies on data from the Primavera Unifier Staging Tables and related publication services. You must configure and run publication services prior to running the STARETL process. This enables users to control what data to calculate and store and how often the data is updated.

> **Note**: For detailed instructions on publishing data in Primavera Unifier, see "Publishing Unifier Data to Analytics" in the *Primavera Unifier Administration Guide*.

To configure publication services in Primavera Unifier, complete the following steps:

1) Sign in to Primavera Unifier with an account that has administration privileges (ootbadmin).
2) Go to the **Company** workspace.
3) Change the mode from user mode to **admin** mode.
4) Open the **Analytics** module.
5) Click the **Schedule** icon 📷 on the toolbar.
6) On the **Schedule Setup** pane, set the data update frequency.
7) If you have made any changes to the data definition or the data mapping fields, select the **Refresh all data** check box.

> **Note**: If you choose to refresh all data, you may want to deselect the **Refresh all data** check box after the run is complete.

8) Click **OK**.
9) If you are using Oracle Data Integrator (ODI), complete the additional steps described in Configuring ODI Source User and Source Schema Privileges.

Creating History Data for Primavera Unifier

Primavera Unifier also provides the ability to capture and store history data. Data can be captured at either the Monthly or Weekly intervals, (with the exception of Space history which is monthly only). No additional settings are required.

Using Location Data

Both P6 EPPM and Primavera Unifier can store location information for projects. In addition, P6 EPPM can also store location information for activities and resources. Primavera Analytics can create intelligent analysis to evaluate measures using the geographical location data entered in P6 EPPM and Primavera Unifier. To accomplish this, you will need spatial boundary data and maps. Primavera Analytics comes with access to Oracle eLocation services. Oracle eLocation service is a cloud-based map data service that provides mapping, geocoding, and routing capabilities. For more information, see the Oracle Business Intelligence (OBI) documentation on mapping and spatial information.

Primavera Analytics now uses built-in Oracle Maps within OBI by default for creating spatial analyses included with Primavera Analytics sample catalog without any external schema or additional Map Viewer configuration. Alternatively, you can continue to use Map Viewer demo data (MVDEMO) for spatial analyses.

> **Note**: If you choose to use Map Viewer demo data (MVDEMO), the location attributes you enter in P6 EPPM and Primavera Unifier must match the attributes in the MVDEMO data to ensure the map layers render correctly.
>
> For example, if you configure your map to use the CITIES_ALL tile, you should query the MVDEMO database user schema and use the exact, case sensitive code that resides in the schema. This means if you have a city name, *Philadelphia,* in P6 EPPM, the name *Philadelphia* must also exist in the MVDEMO CITIES_ALL tile.

In This Section

Entering P6 EPPM Data for Mapping

P6 EPPM supports the ability to enter location data at the dictionary level that can be assigned to Projects, Activity, and Resources and then extracted, transformed, and loaded into Primavera Data Warehouse for use in Oracle Business Intelligence (OBI) analysis.

To create location data in P6 EPPM, complete the following steps:

1) Sign in to P6 EPPM with an account that has the privilege to change enterprise data (for example, admin).
2) From the **Administer** ▼ menu, select **Enterprise Data**.
3) In the **Enterprise Data** pane, expand **Global** and select **Locations**.
4) Enter the location data as appropriate for your specific needs.
5) Assign locations data to projects, activities, and resources as needed.

> **Note**: See the P6 help for information about entering and assigning location data.

6) After all of the location data is entered and assigned where appropriate for your environment, run all of the Global Publication Services to ensure that the data in the publication tables is up to date. After Publication Services have run to successful completion, run the STARETL process to extract, transform, and load the data from P6 EPPM to Primavera Data Warehouse. See the *Primavera Data Warehouse Installation and Configuration Guide* for details.

Using Primavera Unifier Location Data

Preparing Primavera Unifier Data for Mapping

Primavera Unifier supports the ability to enter location data that be assigned to project shells and then extracted, transformed, and loaded into Primavera Data Warehouse for use in Oracle Business Intelligence (OBI) analysis.

To enter location data in Primavera Unifier:

1) Sign in to Primavera Unifier.
2) Assign locations to your projects.
3) From the **Company Admin** screen, select the **Analytics** module.
4) In the Shell mapping configuration, ensure that the **State**, **State Code**, **Country**, and **Country Code** columns are mapped to the appropriate Primavera Unifier data elements.

   - **Note**: When you map a Shell to a UDF in the Unifier Analytics module, the data type of the UDF determines whether it will be available as a dimension or a fact in in analytics.

   - If the data type of the UDF is *String* or *Date*, the data will be mapped as a Dimension object in Primavera Analytics.

   - If the data type of the UDF is *Numeric*, then the data will be mapped as a Fact object in Primavera Analytics, and is only available in the **Primavera - Project User Defined Fields** and **Primavera - Project User Defined Fields History** subject areas.

After you have entered and assigned all of the location data where appropriate for your environment, run all Publication Services to ensure that the data in the publication tables is current. After Publication Services have run successfully, run the STARETL process to extract, transform, and load the data from Primavera Unifier to Primavera Data Warehouse.

Installing and Configuring MVDEMO Sample Spatial Data

Map Viewer demo data (MVDEMO) is an optional installation component for rendering spatial analyses. This section describes how to:

▶ set up and configure Sample App MapViewer content
▶ integrate with Primavera Analytics and Oracle Business Intelligence (OBI)

To begin the process:

1) Download the MVDEMO Sample Data Set from
   http://www.oracle.com/technetwork/middleware/mapviewer/downloads/index.html.

2) Follow the instructions in the MVDEMO readme file that came with the MVDEMO dump file
   to import the MVDEMO sample data set.

Importing Modified MVDEMO Themes

Some of the themes within the MVDEMO Sample Data Set are not configured properly for
Oracle Business Intelligence 12c. In order to correct this, you will need to import the modified
themes using Oracle Map Builder. Oracle Map Builder is a standalone application that allows
you to create and manage the mapping metadata (about styles, themes, and base maps) that is
stored in the database. For more information on Oracle Map Builder, see
https://docs.oracle.com/middleware/1213/core/JIMPV/vis_mapbuilder.htm#JIMPV7000.

To import the modified themes for use with OBI 12c, complete the following steps:

1) Open Oracle Map Builder by running the following command:

```
java –jar mapbuilder.jar
```

2) In the top left section of the screen, click on the **Database** drop-down menu, and then select
   Load/Add/Remove.

3) In the **Load/Add/Remove Database Connections** window, click **Add**.

4) In the **Add Connection** form:

   a. In the **Database** drop-down menu, keep **ORACLE** selected.

   b. Enter a name in the **Connection Name** field.

   c. Enter the **User** and **Password** for the MVDEMO schema (for example,
      mvdemo/mvdemo).

   d. On the **Basic** tab, enter the **Host**, **Port**, and **SID** for the MVDEMO database instance.

   e. Click **Test Connection** to verify the connection parameters.

   f. If connection test is successful, click **OK**.

5) Open the **Tools** menu, and select the **Import Metadata** menu item.

6) Click the **File** button, and browse to **<Primavera Analytics unzipped
   directory>\obi\12c\mapbuilder**.

7) Select **PrimaveraAnalyticsThemes.dat** and click **Open**.

8) Check only the **Themes** and **Overwrite metadata that exists** options, ensure all other items
   are unchecked.

9) Click **OK** to import the themes. Once import has completed, exit Oracle Map Builder.

Configuring mapViewerConfig.xml for MVDEMO

Complete the following steps to configure the MapViewer configuration file
(mapViewerConfig.xml) for MVDEMO:

1) Go to `http://<machine>.<domain>:<port>/mapviewer` to access MapViewer.

2) Log in to MapViewer using the same administration user you use to log in to the WebLogic and Enterprise Manager console.

3) Select **Configuration**. The **mapViewerConfig.xml** configuration file opens.

4) Under the **<security_config>** tag, set the following values:

```
<disable_direct_info_request> true </disable_direct_info_request>
<disable_info_request> true </disable_info_request>
<disable_csf>true</disable_csf>
<enforce_security_role> true </enforce_security_role>
```

5) Under the **<security_config>** tag set the machine name. Replace <machine> and <domain> with the host name where the MapViewer instance is running:

```
<proxy_enabled_hosts>
http://<machine>.<domain>:<port>/mapviewer,http://<machine>.<domain>:<port>/
</proxy_enabled_hosts>
```

6) Scroll down to the sample data source (which is commented out):

```
<map_data_source name="mvdemo"
jdbc_host="db1.my_corp.com"
jdbc_sid="orcl"
jdbc_port="1521"
jdbc_user="scott"
jdbc_password="!tiger"
jdbc_mode="thin"
number_of_mappers="3"
allow_jdbc_theme_based_foi="false"
/>
```

7) Use this sample data source to create your own data source to the database where the spatial data and MapViewer metadata resides. Copy and paste the tag just below the existing one. Make sure that you do not copy the comment tags (**<!–** and **–>**). For example:

```
<map_data_source name="mvdemo"
jdbc_host="<machine>.<domain>"
jdbc_sid="<OracleSID>"
jdbc_port="<port>"
jdbc_user="mvdemo"
jdbc_password="!mvdemo"
jdbc_mode="thin"
number_of_mappers="8"
allow_jdbc_theme_based_foi="true"
editable="false"
/>
```

where <machine> and <domain> are the hostname, <OracleSID> is your Oracle SID value (by default, **obiee12c**), and <post> is your database post (by default, **1521**).

Note: The exclamation point (!) preceding the password indicates that

the password will be encrypted when the MapViewer instance is restarted.

8) Click **Save**.

9) Scroll down to the sample Map Data Server configuration (which is commented out). Enter the following as the Map Data Server configuration:

```
<mds_config>
  <data_source_name="mvdemo">
    <allow_predefined_themes>true</allow_predefined_themes>
    <allow_dynamic_themes>true</allow_dynamic_themes>
  </data_source>
</mds_config>
```

10) Click **Save**.

11) After the save completes, click **Restart**.

Restarting OBI Services

If changes are performed that require the Oracle Business Intelligence (OBI) server to be restarted, use the steps in this section to restart the OBI services.

Complete the following steps to restart the OBI services:

1) Log in to the OBI Enterprise Manager URL with an account that is a member of the WebLogic BIAdministrator group (for example, weblogic):

```
http://<server>:<port>/em
```

where `<server>` is the server name or IP address of the server where OBI is installed and running, `<port>` is the listening port number for OBI, and `/em` is the default context for OBI Enterprise Manager.

2) At the top left of the screen, click the **Target Navigation** icon to expand the **Target Navigation** tree. Click the arrow icon next to the leaf node called **Business Intelligence** to expand this folder.

3) With the **Business Intelligence** leaf node expanded, the **biinstance** hyperlink should appear below the **Business Intelligence** leaf node.

4) Click the **biinstance** hyperlink. This will display on the right portion of the screen, the properties and available configuration options for this deployment of OBI.

5) Click the **Availability** tab.

6) From the **Processes** tab, click **Restart All** to restart all **Oracle Business Intelligence** processes.

7) When prompted to confirm, click **Yes**.

8) When the OBI deployment restarts, click the username at the top right of the screen and select **Log Out**.

> **Note**: Restarting the OBI services temporarily disables all functionality within OBI without warning. Therefore, Oracle highly recommends that

all OBI users save their work prior to restarting.

Verifying the MapViewer Connection

Complete the following steps to verify the MapViewer connection:

1) Access the MapViewer by going to the following URL:

   `http://<machine>.<domain>:9704/mapviewer`

2) Log in to MapViewer using the same administration user that you use to log in to the WebLogic and Enterprise Manager consoles.

3) Select the **Admin** link.

4) Select **Data Sources**.

You should see the data source that you just created. You should also see your machine.domain in your configuration if the changes were successfully committed.

Adding the ELOCATION_MAP for MVDEMO

Primavera Analytics uses a base map called E_LOCATION. To add this base map to MVDEMO, complete the following steps:

1) Access the MapViewer:

   `http://<machine>.<domain>:9704/mapviewer`

2) Select the **Admin** link.

3) Log in to MapViewer using the same administration user you use to log in to the WebLogic and Enterprise Manager console.

4) Select **Create Tile Layer**.

5) Select **Oracle Maps** and click **Continue**.

6) In the **Name** field, verify the name is **elocation_map**.

7) In the **Data Source** field, select **mvdemo**.

8) Select **Submit**.

9) Select **Manage Tile Layers** and verify **ELOCATION_MAP** with Data Source **mvdemo** is there.

Sample of Joining Primavera Data Warehouse Data and MVDEMO Data

The following provides an example of joining location data from the Primavera Data Warehouse database to the sample MVDEMO data:

1) Log into **OBI** with an account that has a BI Administrators role.

2) Select **Administration**.

3) Under **Map Data Management**, click **Manage Map Data**.

4) In the **Manage Map Data** page:

   a. Select the **Layers** tab.

   b. Select **Import Layers**.

5) In the **Import Layers** dialog box:

    a. In the **Look in** field, select **mvdemo**.

    b. Select **COUNTRIES_ALL_LABEL_CLEAR** and **US_STATES_BI**.

    c. Click **OK**.

6) Select **COUNTRIES_ALL_LABEL_CLEAR** and click the **Edit layers** icon.

Configuring Layers

1) Log into **OBI** with an account that has a BI Administrator role.

2) Select **Administration**.

3) Under **Map Data Management**, click **Manage Map Data**.

4) In the **Manage Map Data** page select the **Layers** tab.

5) Select **COUNTRIES_ALL_LABEL_CLEAR** and click the **Edit layers** icon.

6) In the **Edit Layer** dialog box, in the **Name** field, select **ALL COUNTRIES**.

7) Add the **Project-Location** BI Key for the **Primavera - Activity** subject area:

    a. In the **BI Key Columns** section, click **+**.

    b. In the **Select Subject Areas** list, select **Primavera – Activity**.

    c. In the **Select BI Key Columns** dialog box, expand **Project**, then **Location**.

    d. Select the **Country Name** and select **Move**.

    e. Click **OK**.

8) Add the **Activity-Location** BI Key for the **Primavera - Activity** subject area:

    a. In the **BI Key Columns** section, click **+**.

    b. In the **Select Subject Areas** list, select **Primavera – Activity**.

    c. In the **Select BI Key Columns** dialog box, expand **Activity,** then **Location**.

    d. Click the **Country Name** and select **Move**.

    e. Click **OK**.

9) Add the **Resource-Location** BI Key for the **Primavera - Activity** subject area:

    a. In the **BI Key Columns** section, click **+**.

    b. In the **Select Subject Areas** list, select **Primavera – Activity**.

    c. In the **Select BI Key Columns** dialog box, expand **Resource,** then **Location**.

    d. Click the **Country Name** and select **Move**.

    e. Click **OK**.

10) Add the **Project Location** BI Key for the **Primavera - Project History** subject area:

    a. In the **BI Key Columns** section, click **+**.

    b. In the **Select Subject Areas** list, select **Primavera – Project History**.

    c. In the **Select BI Key Columns** dialog box, expand **Project,** then **Location**.

    d. Click the **Country Name** and select **Move**.

    e. Click **OK**.

11) Select **US_STATES_BI** and click the **Edit layers** icon.

12) In the **Edit Layer** dialog box:

    a. In the **Name** field, select **US STATES**.

    b. Use the same steps you used to create **ALL COUNTRIES**, but choose **STATE CODE** instead of Country Name for each step.

13) Repeat these steps as needed for each Subject Area for which you want to display location data on a map.

Configuring Background Maps for MVDEMO

After importing the necessary map layers, complete the following steps to associate the background maps with map layers created in the previous section.

1) Log into **OBI** with an account that has BI Administrator role.
2) Click **Administration**.
3) Under **Map Data Management**, click **Manage Map Data**.
4) In the **Manage Map Data** page:
   a. Select the **Background Maps** tab.
   b. Select **Import Background Maps**.
5) In the **Import Background Maps** dialog box:
   a. In the **Look in** field, choose **mvdemo**.
   b. In the **Available Maps** field, choose **ELOCATION_MAP**.
   c. Select **OK**.
6) Select the **ELOCATION_MAP** you added and select **Edit Background Map**.
7) In the **Edit Background Map** dialog box:
   a. In the **Name** field, enter **City Map**.
   b. In the **Interactive BI Layers** section, click **+**.
   c. In the **Add Layers** pop-up, select **ALL CITIES**.
   d. Click **OK**.
   e. Adjust the zoom level as needed.
   f. Click **OK**.
8) Repeat the above steps to add a country, state, and combination country/state/city maps.

> **Notes**:
> - When you create a country map, name it **Country Map** and add the **ALL COUNTRIES** BI layer.
> - When you create a state map, name it **State Map** and add the **US STATES** BI layer.

When you create a country/state/city map, name it **Country State City Map** and add the **ALL COUNTRIES**, **US STATES**, and **ALL CITIES** BI layers.

Using Advanced Analytics

Advanced Analytics provides you with huge volumes of data that you can use to extract information and discover new insights that enable you to make informed predictions in the Oracle database. Advanced analytics uses open source R with a wide variety of algorithms built in the Oracle database that can be used to resolve a wide gamut of business problems.

This section outlines how to setup and use advanced analytics in Primavera Analytics.

In This Section

Downloading Examples

Download examples of advanced analytics as follows:

1) Go to the Oracle edelivery site.
2) Select the OS of your choice.
3) Navigate to `P_R177_A\sample\AdvancedAnalytics` folder and download the examples.

Predicting the Success of a Project: Example

The example in the **..\samples\Advanced Analytics** folder, predicts if a project will be successful or a failure based on past experiences (with a heavier weight towards cost).

The following project characteristics were considered to create a basis if a project is successful:

▸ Cost
▸ Units
▸ Duration

Now, refer to historical projects and run the formula to see planned vs. actual. For example, planned cost versus actual cost to determine if the project cost was above the planned cost. This provides a success or fail score for all historical projects. Weighted costs was the most important feature of the project.

Having determined what projects were successful or failures historically, you can now gather those characteristics for those projects.

For example, Project A has an original budget of $100,000, planned duration of 2,000 hrs, and planned units of 2,000. This project now also has a score as well - 1 or 0. If a new project is introduced into the system, match it as close as possible to the existing projects and determine what the score is on those projects. Apply that score as a prediction to this new project.

Prerequisites for Advanced Analytics Example

Install the following applications to use advanced analytics:

▶ Installing R

R is an open-source language and environment for statistical computing and data visualization, similar to SAS or SPSS, supporting data manipulation and transformations, as well as sophisticated graphical displays. However, R is limited by memory, processing power on the local machine, single threaded, and scalability.

▶ Installing ORE

Oracle R Enterprise integrates R with Oracle Database. By doing so, Oracle R Enterprise (ORE) removes the memory and computational constraints found in R by executing requested R calculations on data in the database, using the database itself as the computational engine. Oracle R Enterprise allows you to further leverage Oracle's engineered systems, like Exadata, Big Data Appliance, and Exalytics, for enterprise-wide analytics, as well as reporting tools like Oracle Business Intelligence Enterprise Edition dashboards and BI Publisher documents.

Setting Up R

After installing R, set up the R script to connect directly to STARUSER and calculates the predicted outcome for pre-selected projects as follows:

1) Create `table etl_aa`, which contains the `project_object_id` for the projects that will be calculated.

```
CREATE TABLE etl_aa(project_object_id number(19));
```

2) Insert the `project_object_id` for the selected project.

```
INSERT INTO etl_aa(project_object_id) values (4351);
INSERT INTO etl_aa(project_object_id) values (4352);
INSERT INTO etl_aa(project_object_id) values (4354);
INSERT INTO etl_aa(project_object_id) values (3577);
INSERT INTO etl_aa(project_object_id) values (4049);
INSERT INTO etl_aa(project_object_id) values (4301);
commit;
```

3) Open R Studio.

   a. Install the following packages: `rJava, RJDBC, glm2`

```
install.packages("rJava")
install.packages("RJDBC")
install.packages("glm2")
```

   b. Set the working directory to a folder that contains `ojdbc6.jar.`

4) Open the attached R script.

   a. Change the database connection.

   b. Run the script.

      The results will be in `W_UDF_PROJECT_D`

Setting Up Oracle R Enterprise (ORE)

After installing Oracle R Enterprise (ORE), set it up for advanced analytics as follows:

1) Grant permissions.

```
% sqlplus / AS SYSDBA
SQL> GRANT RQADMIN to RQUSER;
```

2) Go to P6 and create user with security permissions named rquser. The name of the P6 user needs to match the name used to install ORE.

3) Run ETL to get the row level security working.

4) Create table `etl_aa`, which contains the `project_object_id` for the projects that will be calculated.

```
CREATE TABLE etl_aa(project_object_id number(19));
```

5) Insert the `project_object_id` for the selected project (the data below is an example based on sample data).

```
INSERT INTO etl_aa(project_object_id) values (4351);
INSERT INTO etl_aa(project_object_id) values (4352);
INSERT INTO etl_aa(project_object_id) values (4354);
INSERT INTO etl_aa(project_object_id) values (3577);
INSERT INTO etl_aa(project_object_id) values (4049);
INSERT INTO etl_aa(project_object_id) values (4301);
commit;
```

6) Login as a `SYSTEM` user grant privileges to `RQUSER` to access `STARUSER`.

```
grant select on staruser.w_project_d to rquser;
grant select on staruser.etl_aa to rquser;
```

7) Install R library glm2. It used in the script.

```
bash> ORE
> library(ORE)
> ore.connect(user="RQUSER", password="oracle",
service_name="pdborcl", all=TRUE)
> install.packages("glm2")
```

8) Create an R object (script under `RQUSER`).

```
bash> ORE
> library(ORE)
> ore.connect(user="RQUSER", password="oracle",
service_name="pdborcl", all=TRUE)
> ore.scriptCreate(predict_projects ........................ <the
full command is the attached file >
```

9) Login as staruser and run the script:

```
SELECT * FROM table(rqEval(NULL, NULL, 'predict_projects'));
```

10) The prediction results will be stored in `RQUSER.ETL_AA_RESULTS`.

11) Bring `RQUSER.ETL_AA_RESULTS` to `STARUSER`:

   a. Create a database link from `STARUSER` to `RQUSER`:

```
CREATE DATABASE LINK DBLINKR
CONNECT TO RQUSER IDENTIFIED BY oracle
```

```
     USING 'pdborcl';
```

12) Bring over the entire `ETL_AA_RESULTS` table:

```
create table etl_aa_results as
       select * from etl_aa_results@dblinkr;
```

Additional Administrative Tasks

This chapter describes additional administrative tasks, including how to clear and refresh data, how to filter data for Primavera Data Warehouse, and how to schedule or manually launch a database update.

In This Section

Configuring the RPD File for Financial Periods

If you configured additional Financial Periods for use with Primavera Data Warehouse when you installed Primavera Data Warehouse, some additional steps are required to make these additional Financial Periods available in Oracle Business Intelligence (OBI).

To configure OBI to use the additional Financial Periods, complete the following steps:

1) Launch the OBI Administration application.
2) In the OBI Administration application, select **File**, **Open**, **Offline**.
3) Select **OraclePrimaveraAnalytics.rpd** and click **Open**.
4) Enter the password for the OraclePrimaveraAnalytics.rpd file (by default, **Admin123**).
5) In the far right section (**Physical**), go to the **W_DAY_D** table.
6) Select the **FIN_PERIOD_2** and **FIN_PERIOD_3** fields, drag them to the middle section (**Business Model and Mapping**), and release them in the **Dim – Date** dimension.
7) After **FIN_PERIOD_2** and **FIN_PERIOD_3** appear in the **Business Model and Mapping** section under the **Dim – Date** dimension, you can drag them to the left section (**Presentation**) where you want them to appear in OBI. You can also rename them there, so they display as you require.
8) Save **OraclePrimaveraAnalytics.rpd**.

Setting Up the Role Utilization Subject Area for P6 EPPM 15 R2

Beginning with version 16 R1, the Role Utilization subject area is included in Primavera Analytics. The underlying views for this subject did not exist in the P6 EPPM Extended Schema (Px) before P6 EPPM 16 R1. However, if you are using a P6 EPPM 15 R2 database with Primavera Analytics 16 R1, you can manually create the views required for the Role Utilization subject area.

> **Note**: The Role Utilization subject area is not supported with P6 EPPM 15 R1 or lower.

To do this when using P6 EPPM with an Oracle database, complete the following steps:

1) Go to **<Primavera Analytics unzipped directory>\px_views**.
2) Open **OR_px_role_limit_views.sql** in a text editor.
3) Log in to the P6 EPPM database instance (using SQL Plus, SQL Developer, or a similar tool) with the P6 EPPM Extended Schema user (for example, PXRPTUSER).
4) Run the create view statements in **OR_px_role_limit_views.sql** to create the two views required to support the Role Utilization subject area (ROLELIMIT and ROLEAVAILABILITY).

To do this when using P6 EPPM with a Microsoft SQL Server database, complete the following steps:

1) Go to **<Primavera Analytics unzipped directory>\px_views**.
2) Open **SS_px_role_limit_views.sql** in a text editor.
3) Log in to the P6 EPPM database instance (using SQL Plus, SQL Developer, or a similar tool) with the P6 EPPM Extended Schema user (for example, PXRPTUSER).
4) Run the create view statements in **SS_px_role_limit_views.sql** to create the two views required to support the Role Utilization subject area (ROLELIMIT and ROLEAVAILABILITY).

Clearing the Data from Primavera Data Warehouse

> **Caution**: Clearing the data from the Primavera Data Warehouse database removes all data, including all accumulated Project, Activity, and WBS history fact data. You cannot "undo" this action.

If needed, complete the following steps to drop all of the data (including all accumulated Project, Activity, and WBS history fact data) from the Primavera Data Warehouse schema:

1) Log in to the Primavera Data Warehouse instance (with SQL Plus, SQL Developer, and so on) with a user account that has the Drop User privileges (for example, SYSTEM).
2) Run the following command:

```
drop user <staruser> cascade;
```

where <staruser> is the username for your Primavera Data Warehouse (by default, STARUSER).

You can run the Setup process to populate the Primavera Data Warehouse database after executing this command. See the *Primavera Data Warehouse Installation and Configuration Guide* for details on running the Setup process.

Filtering Data for Primavera Data Warehouse

You can apply a filter during the STARETL process that can limit the data loaded into Primavera Data Warehouse based on a specific criteria. By creating a filtered view, you can limit the projects you include in Primavera Data Warehouse to a portfolio, an EPS, or any other specified criteria. In this way, you can customize Primavera Data Warehouse with only the projects that are needed.

If you want to create a filtered source of data for loading into Primavera Data Warehouse, the filter must be created with the user account that was created for the publication tables/views (for example, PXRPTUSER) with the P6 EPPM database instance.

Before creating a new filtered view, Oracle recommends that a new data source be added to Primavera Data Warehouse. Doing this ensures that the existing STARETL process and associated data is unaffected by any filtering that is occurring. To create a new data source, see the *Primavera Data Warehouse Installation and Configuration Guide* for details on how to create and set up a new data source for details.

Filtering P6 EPPM Data

The following example demonstrates how to create a new filtered view (called **My Top 10 Projects**), name the filter (**it_portfolio_view**) and how to tell the STARETL process to use this new filtered view when extracting, transforming, and loading data into Primavera Data Warehouse.

1) Log in to the P6 EPPM instance (with SQL Plus, SQL Developer, and so on) with the user account for Publication Tables/views (for example, PXRPTUSER).
2) Use the following statement to create the **it_portfolio_view** filter:

```
create or replace view it_portfolio_view as select PROJECTOBJECTID objected from
projectportfolio pp, projectprojectportfolio ppp where pp.objectid =
ppp.PROJECTPORTFOLIOOBJECTID and pp.name = 'My Top 10 Projects'
```

3) Upon successful completion of the sample filter, log out of the instance.
4) Log in to the Primavera Data Warehouse database instance as the schema owner (for example, the STARUSER account).
5) Use the following statement to add a setting into the ETL_PARAMETER table with the name of the view you created:

```
insert into etl_parameter(p_feature, p_1, datasource_id)
values ('star.project.filter.ds1', 'it_portfolio_view', 1);
commit;
```

6) Go to **<PDW Install Folder>\star\etl_homes\staretl<id>**.
7) Run **staretl.bat** (with Windows) or **staretl.sh** (with UNIX or Linux).

Filtering Primavera Unifier Data

The following example demonstrates how to create a view for only active projects and a filter called **active_projects**.

1) Log in to the Primavera Unifier instance (with SQL Plus, SQL Developer, and so on), with the schema owner account (for example, unifier).

2) Use the following statement to create the **active_projects** filter:

```
create or replace view active_projects as select project_object_id objected from
sys_stg_project where project_status = 'Active'
```

3) Upon successful completion of the sample filter, log out of the instance.

4) Log in to the Primavera Data Warehouse database instance as the schema owner (for example, the STARUSER account).

5) Use the following statement to add a setting into the ETL_PARAMETER table with the name of the view you created:

```
insert into etl_parameter(p_feature, p_1, datasource_id)
values ('star.project.filter.ds1', 'active_projects', 1);
commit;
```

6) Go to **<PDW Install Folder>\star\etl_homes\staretl<id>**.

7) Run **staretl.bat** (with Windows) or **staretl.sh** (with UNIX or Linux).

Scheduling STARETL Process Runs

To keep the Primavera Data Warehouse database current, run the STARETL process periodically. Typically, you will run the STARETL process nightly during off-hours.

You can use Task Scheduler (with Windows) or a cron job (with UNIX or Linux) to schedule STARETL process runs. Refer to your operating system documentation for specifics on how to configure the process to run as a recurring job.

You can also use the web-based Configuration Utility to schedule STARETL process runs. See the *Primavera Data Warehouse Installation and Configuration Guide* for details.

> **Caution:** When scheduling STARETL process runs, Oracle recommends that you do not start a STARETL process run until the previous run has completed.

Backing up the Primavera Data Warehouse Database

Oracle strongly recommends that you create nightly backups of the Primavera Data Warehouse database to preserve historic data and ensure you have a steady environment in case a problem occurs.

See http://docs.oracle.com/cd/E11882_01/backup.112/e10642/rcmintro.htm#BRADV8001 for details on backup and recovery.

Comparing Repository Definition (.rpd) Files

Oracle Business Intelligence (OBI) provides the ability to generate an XML file based on the differences between two Repository Definition (.rpd) files. This can be useful to track any changes in Primavera Analytics Repository Definition (.rpd) files from release to release. Additionally, the XML difference file that this utility generates can be used to apply new changes to an existing OBI Repository Definition (.rpd) file. See https://docs.oracle.com/cd/E14571_01/bi.1111/e10540/mngreposfiles.htm#BIEMG638 for details on this utility.

Creating an XML Difference File

To generate an XML Difference File that contains the differences between two Repository Definition (.rpd) files, complete the following steps:

> **Note**: Do not complete these steps if you plan to merge the two files. For details, see *Merging Repository Definition (.rpd) Files* (on page 48).

1) Copy the two Repository Definition files you want to compare (for example, the currently deployed .rpd file and Primavera Analytics 15 R2 .rpd file) into a new folder.
2) Launch the OBI Administration application.
3) Select **File**, **Open**, **Offline**.
4) From the Open submenu, select the new.rpd file (that is, the .rpd file that contains the updates that you want to apply to the original .rpd file).
5) After the new file opens, in the BI Administration tool, select **File**, **Compare**.
6) In the **Compare repositories** window, click **select, repository** to select the original .rpd file (the copy of the .rpd file that is currently deployed).
7) Choose **No** when asked if you want to mark differences in the repository immediately.

> **Note**: The process compares the two repositories. The Compare repositories window shows a list of differences between the two repositories.

8) Click **Create Patch** to create an XML file that contains these identified differences.
9) In the **Create Patch** window:
   a. Choose a location to save the XML file.
   b. Create a name for the differences XML file.
   c. Leave the **Save as type** option set to **XML files**.

A new XML file is created that contains the differences between the two Repository Definition files. You can open this file in any editor that supports an XML formatted file. This XML file can be applied to an existing repository definition file to merge changes from one repository definition file into another.

Applying the XML Difference File to an Existing Repository Definition File

After creating an XML Difference File and reviewing the it, you can apply the XML Difference File to an existing Repository Definition file.

To apply an XML Difference File to an existing OBI Repository Definition file, complete the following steps:

1) Launch the OBI Administration application.
2) Select **File**, **Open**, **Offline**.
3) From the Open submenu, choose the original .rpd file (the copy of the .rpd file that is currently deployed).
4) After the original file is open in OBI Administration, select **File**, **Merge**.
5) In the **Merge Repository** wizard:
   ‣ **Merge Type**: Patch Repository Merge
   ‣ **Original Master Repository**: The copy of the original .rpd file.
   ‣ **Repository Password**: The password for the original .rpd file (by default, **Admin123**).
   ‣ **Patch File**: The file saved above.
   ‣ **Save Merged Repository as**: A new file name and location which represents the repository definition file that contains the merged repository.
   ‣ **Repository Password**: The password for the newly created merged .rpd file   (by default, **Admin123**).
6) Click **Next.**

> **Note**: This process may take a few minutes while the wizard processes the XML file and creates the new merged .rpd file.

7) On the **Define Merge Strategy** window, decide how to apply the differences in the XML file to the original .rpd file. For details about this process, see http://docs.oracle.com/cd/E28271_01/fusionapps.1111/e20836/merge.htm.
8) Click **Finish** to create the merged .rpd file.

Merging Repository Definition (.rpd) Files

Oracle Business Intelligence (OBI) has the ability to merge two Repository Definition (.rpd) files together to form one file. This method is an alternative to creating a differences XML file, verifying the changes, then merging the changes together.

> **Note**: Do not complete these steps if you plan to update the definition (.rpd) file by creating and applying an XML Difference file. See *STARETL Process Calculations for P6 EPPM Data* (on page 51) for details.

Creating a Merged Repository Definition (.rpd) File

To merge two repository definition (.rpd) files, complete the following steps:

1) Launch the OBI Administration application.

2) Select **File**, **Open**, **Offline**.

3) From the open submenu, choose the original .rpd file (the copy of the .rpd file that is currently deployed).

4) After the original file is open in OBI Administration, select **File**, **Merge**.

5) In the **Merge Repository** wizard:

   ‣ **Merge Type**: Patch Repository Merge

   ‣ **Original Master Repository**: The copy of the original .rpd file.

   ‣ **Repository Password**: The password for the original .rpd file (by default, **Admin123**).

   ‣ **Patch File**: The file saved above.

   ‣ **Save Merged Repository as**: A new file name and location which represents the repository definition file that contains the merged repository.

   ‣ **Repository Password**: The password for the newly created merged .rpd file (by default, **Admin123**).

6) Click **Next.**

> **Note**: This process may take a few minutes while the wizard processes the XML file and creates the new merged .rpd file.

7) If you moved the .rpd to a new OBI installation, do the following:

> **Note**: If you did not move the .rpd to a new OBI installation, skip these steps.

   a. Copy the **OraclePrimaveraAnalytics** file to the **repository** folder on the server where you installed OBI.

   Example:
   ```
   $ORACLE_INSTANCE/instances/instance1/bifoundation/OracleBIServerCo
   mponent/coreapplication_obips1/repository
   ```

   b. Log in to the OBI Enterprise Manager URL with an account that is a member of the WebLogic BIAdministrator group (for example, weblogic):

   ```
   http://<server>:<port>/em
   ```

   where `<server>` is the server name or IP address of the server where OBI is installed and running, `<port>` is the listening port number for OBI, and `/em` is the default context for OBI Enterprise Manager.

   c. Upon logging into OBI Enterprise Manager, the left side of the screen will show the Target Navigation Tree. Click the arrow icon next to the leaf node called **Business Intelligence** to expand this folder.

   d. With the **Business Intelligence** leaf node expanded, the **coreapplication** hyperlink should appear below the **Business Intelligence** leaf node.

   e. Click the **coreapplication** hyperlink. This will display on the right portion of the screen, the properties and available configuration options for this deployment of OBI.

f. The right side of the screen is organized in a parent/child tab-based display. This means for each parent tab that has the focus, a set of child tabs will be available. Click the **Deployment** parent tab, then click the **Repository** child tab.

g. In the **Change Center** section above the tabs, click the **Lock and Edit Configuration** link. Clicking this link allows the fields in this deployment to be edited. A message appears confirming that the lock and edit command completed successfully. Click **Close** to dismiss this message.

h. In the **Upload BI Server Repository** section, click **Browse**.

i. Go to the directory that contains the modified **OraclePrimaveraAnalytics.rpd** file.

j. Select the modified **OraclePrimaveraAnalytics.rpd** file and click **Open**.

k. In the **Repository Password** field, enter the **OraclePrimaveraAnalytics.rpd** password (by default, **Admin123**).

l. In the **Confirm Password** field, re-enter the **OraclePrimaveraAnalytics.rpd** password.

m. At the upper right of the **Repository** child tab, click **Apply**.

> **Note**: You can configure and deploy the Primavera Analytics sample catalog prior to activating the changes and restarting the OBI services. See the *Analytics Installation and Configuration Guide* for details.

n. In the **Change Center** section above the tabs, click the **Activate Changes** link. A confirmation message appears. Click **Close to dismiss the** confirmation window.

o. In the **Change Center** section above the tabs, click the **Restart To Apply Recent Changes** link. Clicking this link will go to the **Overview** parent tab for this OBI deployment.

p. In the **System Shutdown & Startup** section, directly below the Pie Chart graph showing system components status, click **Restart** to restart all components of this OBI deployment.

q. When the OBI deployment has been restarted, a System Restarted Successfully message displays. Click

r. **Close** to close the message.

s. Click the **Logout** hyperlink (in the upper right of the screen directly below the browser toolbar) to log out of OBI Enterprise Manager.

Deploying the Newly Merged Repository Definition (.rpd) File

You can now deploy the newly merged Repository Definition (.rpd) file in Oracle Business Intelligence (OBI). To deploy the newly merged Repository Definition (.rpd) file, complete the following steps:

1) Open a terminal or command prompt window and go to your BI Domain tools folder. For example:

```
C:\Oracle\Middleware\Oracle_Home\user_projects\domains\bi\bitools\bin
```

2) Use the **data-model-cmd.cmd** command (with Windows) or the **data-model-cmd.sh** command (with UNIX or Linux) in the terminal or command prompt to upload the newly merged Repository Definition file using your WebLogic administrator credentials (the -U and –P parameters). For example:

```
data-model-cmd.cmd uploadrpd –I C:\Temp\OraclePrimaveraAnalytics162.rpd –W
Admin123 –U weblogic –P weblogic1 –SI ssi
```

   You should receive a message indicating that the RPD Upload completed successfully.

3) Log in to the OBI Enterprise Manager URL with an account that is a member of the WebLogic Administrator group (for example, weblogic):

```
http://<server>:<port>/em
```

   where `<server>` is the server name or IP address of the server where OBI is installed and running, `<port>` is the listening port number for OBI, and `/em` is the default context for OBI Enterprise Manager.

4) Click the Target Navigation Tree icon at the upper left corner of the scree to expand it.

5) Click the arrow icon next to the Business Intelligence leaf node to expand the folder.

6) Click the **biinstance** hyperlink, which appears below the leaf node, to display the properties and available configuration options for this deployment of OBI on the right side of the screen.

7) Click the **Availability** parent tab

8) Click the **Processes** child tab.

9) Click the **Restart All** link to restart all components of this OBI deployment.

10) When the confirmation window appears, click **Yes**.

    When the OBI deployment restarts, a confirmation message appears.

11) Click **Close**.

12) Click the user menu (for example, weblogic) located at the upper right of the screen directly below the browser toolbar, and select **Log Out**.

STARETL Process Calculations for P6 EPPM Data

Included with the STARETL process, some fields are calculated and stored in Primavera Data Warehouse based on predefined calculations. These calculations can be changed to meet specific business requirements.

To view the predefined calculations, complete the following steps:

1) Log in to Primavera Data Warehouse (with SQL Plus, SQL Developer, and so on) with the user account for the schema owner (for example, STARUSER).

2) Run the following command:

```
Select * from etl_calculations;
```

The output from this sql statement shows the following columns:

▸ **TABLE_NAME**: This is the table name where the calculation will be stored

- **COLUMN_NAME**: This is the column name for the calculation
- **CALCULATION**: This is the actual calculation which is used to create the data in the column
- **DEFAULT CALCULATION**: This field shows what the calculation for this column is/was for the default out-of-the box calculation.
- **DATA SOURCE ID**: This is the data source id for the calculation.

The following Primavera Data Warehouse can have calculations:

- **W_ACTIVITY_HISTORY_F**
- **W_WBS_HISTORY_F**
- **W_PROJECT_HISTORY_F**
- **W_ACTIVITY_WORK_DOWN_HF**
- **W_ASSIGNMENT_WORK_DOWN_HF**

> **Note**: You cannot add additional rows to this ETL_CALCULATIONS table.

Changing a STARETL Process Default Calculation

This section provides an example of how to change a STARETL process default calculation. Each row in the ETL_CALCULATIONS table represents a calculation that can be changed. In order to change a calculation, you must be logged on to the Primavera Data Warehouse instance with the schema owner account (for example, STARUSER). Additionally, you must use a database update statement followed by a commit.

The following example shows how to update the default calculation for a row called **EMERGENT_WORK**:

1) Log in to Primavera Data Warehouse (with SQL Plus, SQL Developer, and so on) with the user account for the schema owner (for example, STARUSER).
2) Run the following command:

```
update etl_calculations set calculation = 'CASE WHEN a.actual_start_date >=
s.period_start_date and a.actual_start_date <= s.period_end_date THEN 1 ELSE 0
END' where column_name = 'EMERGENT_WORK' and TABLE_NAME='W_ACTIVITY_HISTORY_F';
commit: /
```

Validating Changed STARETL Process Calculations

When updating metadata calculations, the STARETL process validates the calculations. These validation steps check the etl_calculations table, CALCULATION column, and row values to ensure that the set calculation will execute successfully and not cause the STARETL process to fail. If a calculation causes a failure, all metadata calculations revert to the default calculations and the STARETL process continues. The log reports any failure of the calculation and gives a warning that the calculations will be reverted.

You can run this validation process independently of the STARETL process. Validating changed calculations ensures that any changed calculations are correct. Since the full STARETL process is not run, all calculations are not reverted to default values.

Complete the following steps to validate any changes to STARETL process calculations:

1) Go to **<PDW Install Folder>\star\etl_homes\staretl<x>**.

2) Run one of the following:

   `staretl.bat 10 12` (with Windows)

   `staretl.sh –from 10 –to 12` (with UNIX or Linux)

   > **Note**: Steps 10 through 12 in the STARETL process validate calculations. These commands run these specific steps only.

3) Check the log output for any errors or warnings. See the *Primavera Data Warehouse Installation and Configuration Guide* for details on what logs to check and the log file locations.

   > **Tips:**
   >
   > ▪ Calculations are defined by the Primavera Data Warehouse schema, not by each data source. If you have multiple data sources loading data into a single Primavera Data Warehouse schema, all data sources share the same calculations.
   >
   > ▪ The default calculations are stored in the **ETL_CALCULATIONS** table in the **DEFAULT_CALCULATION** column. You can use the **DEFAULT_CALCUALTION** column to reset a changed calculation back to the original value.

Syncing Cloud and On-Premises Data

Use the **Primavera Analytics Cloud Sync** utility to populate your on-premises Primavera Data Warehouse with data from cloud.

> **Note**: To use **Primavera Analytics Cloud Sync**, you must have a license for Primavera Analytics, on-premises or Primavera Data Warehouse on-premises software.

In This Section

Syncing P6 EPPM Data from Cloud to On-Premises

Complete the following sequence to sync P6 EPPM cloud data (source) to your on-premises (target) data:

1) *Prerequisites to Sync P6 Cloud and On-Premises Data* (on page 55)
2) *Configuring the Sync Environment for P6 EPPM Cloud to On-Premises for Primavera Data Warehouse* (on page 56)
3) Configuring and Running P6 EPPM Publication Services from Analytics Web Application

Prerequisites to Sync P6 Cloud and On-Premises Data

Before proceeding with the sync setup, choose one of the following:

▸ Initial Sync
▸ Incremental Sync after Dump Restoration

Initial Sync is recommended for only small databases where number of Projects is less than 100. Otherwise, choose Database dump restoration and Incremental Sync.

Initial P6 Sync Prerequisites

1) Create an empty P6 database (Target) on an Oracle 12c database instance.
2) Contact Oracle Support to:
   ▸ Request Server sync to be configured in the cloud and get details of URL to access WebLogic Sync server and its authentication information. Also get the property values for **sync.name** and **sync.number** given in the **client_send.properties** file while configuring the server.
   ▸ Ensure that the **optin.project_mode** property is set to '*PX*' in the **client_send.properties** file.
3) Set JAVA_HOME to the supported version. For more details see *client Requirements* document.

Database Dump Prerequisites

1) Contact Oracle Support to request for backup dump file of your source P6 database.

2) Import the provided P6 source admuser database dumpfile in P6 Target database (on-premises).

3) Contact Oracle Support:

   ‣ Request Server sync to be configured in the cloud and get details of URL to access Weblogic Sync server and its authentication information. Also get the property values for **sync.name** and **sync.number** given in the **client_send.properties** file while configuring server.

   ‣ Ensure that the **optin.project_mode** property is set to *INCRPX'* in the **client_send.properties** file.

4) Set the JAVA_HOME to the supported version. For more details see *client Requirements* document.

Configuring the Sync Environment for P6 EPPM Cloud to On-Premises for Primavera Data Warehouse

To receive P6 EPPM cloud data: :

1) Contact Oracle Support with a service request to enable **Primavera Analytics Cloud Sync**.

> **Note**: To use **Primavera Analytics Cloud Sync, y**ou must have a license for Primavera Analytics on-premises or Primavera Data Warehouse on-premises software.

2) When the service request is completed, download **sync-client.zip** from the Primavera Portal landing page.

3) Extract **sync-client.zip**.

4) (Initial sync only) From the **sync-client** directory run the **clear_all_p6.sql** script against the target P6 admuser schema.

5) For Unix or Linux systems, update permissions on .sh files using the following command.
   ```
   $ chmod u+x *.sh
   ```

6) To receive data from the cloud, edit **client_receive.properties** in the **sync-client** directory with the following information:

   ‣ **sync mode =** *CLIENT* This information is pre-populated.

   ‣ **sync type =** *RECEIVER* This information is pre-populated.

   ‣ **sync.name**=<Name of the sync> A unique sync name pre-populated to identify the sync configuration. For example, **p6clouddata_receiver**.

   ‣ **sync.number**=This number is pre-populated.

   ‣ **target.db.host**=The host name (or IP) of the machine on which the P6 target database is hosted. For example, *localhost*.

   ‣ **target.db.port**=The port number for the Oracle instance on which P6 target database is created. For example, 1521.

   ‣ **target.db.service**=The service name of the Oracle instance on which P6 target database is created. For example, *pdborcl*.

> ‣ **target.db.type**=P*6.* .
> ‣ **target.db.schema**=Database username of the admin user (*admuser*) of P6 target database.
> ‣ **target.db.syncuser**=The username of **sync-client** database schema that will be created on same Oracle instance as the P6 target database. It should not be the name of an existing schema, as this schema will be created during setup. This schema user must contain an underscore ("_") character. For example, *t_syncuser*.
> ‣ **target.db.sysuser**=The name of the sysdba user. For example, *sys*.
> ‣ **server.base_url**=The URL of the WebLogic server sync rest service received from Cloud support. This URL is pre-populated. For example, *http://<hostname>:<port#>/sync/rest.*
> ‣ **server.connectuser**= The username of a user who has permissions to create sync configurations. For example, *weblogic* or *admin*. This user is pre-populated.

7) Save **client_receive.properties**.

8) Run **syncsetup.cmd** or **syncsetup.sh** to create and configure the database schema for the sync process.

    ```
    $ ./syncsetup.sh client_receive.properties
    ```

    a. Enter passwords for the following users:

        ▪ syncuser (**target.db.syncuser**)
        ▪ The sysdba user on that same Oracle instance (**target.db.sysuser**)
        ▪ The admin user for the P6 target database schema (**target.db.schema**)
        ▪ The web user who will insert the configuration into the sync configuration (**server.connectuser**)

    The process creates the **t_syncuser** database schema (**target.db.syncuser**).

    b. After the process completes, enter the password of *syncadmin* user. This is the user credential for local Jetty server. Subsequent processes will use this credential to communicate with the server.

9) Start the sync client server.

    ```
    $ . /startserver .sh
    ```

    Server startup will take up to a minute. If you load the configuration without starting the server, you will get an error.

10) Load the sync and receive the configuration into the sync client server:

    ```
    $ . /commander .sh LOAD client_receive.properties
    ```

11) Re-enter the passwords for P6 database admin user (**source.db.schema**) and server connection user (**server.connectuser**).

    The sync client should now start syncing with the sync microservice.

12) To check if the sync process is running, use any of the following options:

> ‣ See **server.log** in the **sync-client/log** folder.
> ‣ Connect as T_SYNCUSER, and check the sync progress from the database. Run the following scripts:
>
>   ▪ To see details of the file posted successfully to the database, run:
>            ```
>            select * from sync_config;
>            ```

- To get an update of the global tables being synced and the status of the sync process for each specific table, run:

  ```
  select * from sync_control;
  ```

  The **entry_value** column status changes from *New* to *Complete*. No records are present in this table as the target schema details are on the client side.

- To display all the projects registered to be synced, run:

  ```
  select * from sync_whitelist;
  ```

  Synced projects will have a **Status** of *Sent* and projects waiting to be synced will have a **Status** of *New*. No records are present in this table as the target schema details are on the client side.

- To view the blob records corresponding to source admuser schema:

  ```
  select * from sync_transfer_data;
  ```

  The status of the record changes from *Receiving, Received, Applying,* and finally to *Applied*. After the record status is *Applied*, check the corresponding data in target admuser schema.

13) To load cloud sync data to Primavera Data Warehouse on-premises database:

   a. Publish the data into P6 tables by:

   - Configuring and Running P6 EPPM Publication Services from P6

     or

   - Configuring and Running P6 EPPM Publication Services from Analytics Web Application

   b. Run the ETL to load the data into your on-premises Primavera Data Warehouse database.

Publishing P6 Data to Extended Schema

To publish P6 data to the extended schema:

1) Get the **p6services.jar** file from the P6 build.

2) Run the following command:

   ```
   java -jar "Path to p6services.jar"
   ```

3) Select *Oracle* or *Microsoft SQL Server* from the **Database Type** list.

4) Enter the following database details: database username, publication user username, password, and database service.

5) Select **Next**.

6) In the P6 target database, run the following as ADMUSER:

   a. Ensure that `last_update_date` is null for all projects that are considered for publishing (**PX_ENABLE_PUBLICATION_FLAG = 'Y'**). If not, update them as follows:

   ```
   update project set px_last_update_date = null WHERE
   PX_ENABLE_PUBLICATION_FLAG = 'Y'; commit;
   ```

   b. Ensure that `setting_value` has the maximum Project ID for **setting_name= 'ProjectNonDemandStopKey'.**

```
update SETTINGS set Setting_value = (select max(proj_id) from project)
where Namespace = 'PxService.Project.Arbiter' and Setting_name =
'ProjectNonDemandStopKey'; commit;
```

7) Ensure that the P6 target database, to which data is synced, is added as a source to the Primavera Data Warehouse.

8) To enable Publication Services view on Webapp, run the following query in the STAR database as STARUSER:

```
UPDATE ETL_PARAMETER SET P_1='true' WHERE P_FEATURE='db.pmdb.cloud';
commit;
```

Syncing Unifier Data from Cloud to On-Premises

Complete the following sequence to sync Unifier cloud data (source) to your on-premises (target) data:

1) *Prerequisites to Sync Unifier Cloud and On-Premises Data*
2) *Configuring Sync Environment for Unifier Cloud and On-Premises*

Prerequisites to Sync Unifier Cloud and On-Premises Data

1) Download **sync-client.zip** from the Primavera Portal landing page.
2) Create an empty Unifier database (target) on an Oracle 12c database instance.

   If there is no existing target database, create an empty Unifier database using the **unifier_sys_stg_tables.sql** following the instructions provided in the SQL script. The **unifier_sys_stg_tables.sql** is located in **sync_client.zip**.
3) Connect to the target Unifier schema with sys user and run the following command

   ```
   grant select on sys.GV_$TRANSACTION to <Target unifier schema>;
   ```
4) Run **auditing_schema_changes_unifier.sql** in the Unifier target schema. The script is located in **sync_client.zip**.
5) Contact Cloud Support to:
   ▸ Request server sync to be configured in the cloud.
   ▸ Get details of the URL to access Weblogic sync server and it's authentication information.
   ▸ Get the property values for **sync.name** and **sync.number** given in **client_send.properties** while configuring server.
6) Set JAVA_HOME to Java 1.8.

Configuring Sync Environment for Unifier Cloud and On-Premises

To receive data from the cloud to an on-premises database, setup and configure **client_receive.properties** as follows:

1) Extract **sync-client.zip**.

   ```
   $ unzip sync_client.zip
   $ cd sync_client
   ```
2) If the target Unifier database exists and is not empty, then run **clear_all_unifier.sql** from current directory against the target Unifier schema.

3) Unzip **sync-client.zip** available in the present working directory and enter the resulting sync-client directory.

```
$ unzip sync-client.zip
$ cd sync-client
```

4) If on a Unix or Linux system, set the executable flag on the shell scripts.

```
$ chmod u+x *.sh
```

5) To receive data from the cloud to an on-premises database, edit **client_receive.properties** as follows:

▸ **sync mode**: *CLIENT* This information is pre-populated.

▸ **sync type**:   *RECEIVER* This information is pre-populated.

▸ **sync.name**: A unique sync name to identify the sync configuration. Obtain this information when you contact Oracle Support. For more details, see ***Prerequisites to Sync Unifier Cloud and On-Premises Data*** (on page 59)..

▸ **sync.number**: The number of the sync configured in the sync server. Obtain this information when you contact Oracle Support. For more details, see ***Prerequisites to Sync Unifier Cloud and On-Premises Data*** (on page 59).

▸ **target.db.host**: The host name (or IP) of the machine on which the Unifier target database is hosted. For example, *localhost*.

▸ **target.db.port**: The port number for the Oracle instance on which Unifier target database is created. For example, *1521*.

▸ **target.db.service**: The service name of the Oracle instance on which Unifier target database is created. For example, *pdborcl*.

▸ **target.db.type**=*UNIFIER*

▸ **target.db.schema**=Database username of the Unifier target database.

▸ **target.db.syncuser**=The username of the **sync-client** database schema that will be created on same Oracle instance as the Unifier target database. It should not be the name of an existing schema, as this schema will be created during setup. This schema user must contain an underscore ("_") character. For example, *t_syncuser*.

▸ **target.db.sysuser**=The name of the sysdba user. For example, *SYS*.

▸ **server.base_url**=The URL of the WebLogic server sync rest service received from Cloud support. Obtain this information when you contact Oracle Support. For more details, see ***Prerequisites to Sync Unifier Cloud and On-Premises Data*** (on page 59). For example, *http://<hostname>:<port#>/sync/rest-service*.

▸ **server.connectuser**= The user name who has permissions to create sync configurations. For example, *weblogic* or *admin*. Obtain this information when you contact Oracle Support.

6) Save **client_receive.properties**.

7) Run **syncsetup.cmd** or **syncsetup.sh** to create and configure the database schema for the sync process:

```
$ ./syncsetup.sh client_receive.properties
```

a. Enter passwords for the following users:

▪ syncuser (**target.db.syncuser**)

▪ The sysdba user on that same Oracle instance (**target.db.sysuser**)

- The admin user for the Unifier target database schema (**target.db.schema**)
- The web user who will insert the configuration into the sync configuration (**server.connectuser**)

The process creates the **t_syncuser** database schema (**target.db.syncuser**).

b. After the process completes, enter the password of *syncadmin* user. This is the user credential for local Jetty server. Subsequent processes will use this credential to communicate with the server.

8) Start the sync client server.

```
$ . /startserver.sh
```

> **Note**: Server startup will take up to a minute. If you load the configuration without starting the server, you will get an error.

9) Load the sync and receive the configuration into the sync client server.

```
$./commander.sh LOAD client_receive.properties
```

10) Re-enter the passwords for Unifier database user (**source.db.schema**) and server connection user (**server.connectuser**).

The sync client should now start syncing with the sync microservice.

11) To check if the sync process is running, use any of the following options:

▸ See **server.log** in the **sync-client/log** folder or

▸ Connect as T_SYNCUSER, and check the sync progress from the database.

- To see details of the file posted successfully to the database, run:

```
select * from sync_config;
```

- To get an update of the global tables being synced and the status of the sync process for each specific table, run:

```
select * from sync_control;
```

The **entry_value** column status changes from *New* to *Complete*. No records are present in this table as the target schema details are on the client side.

- To display all the projects registered to be synced, run:

```
select * from sync_whitelist;
```

Synced projects will have a **Status** of *Sent* and projects waiting to be synced will have a **Status** of *New*. No records are present in this table as the target schema details are on the client side.

- To view the blob records corresponding to the source schema, run:

```
select * from sync_transfer_data;
```

The status of the record changes from *Receiving, Received, Applying,* and finally to *Applied*. After the record status is *Applied*, check the corresponding data in target schema.