

HP Project and Portfolio Management Center

Software Version: Content Pack 2.0

Operational Reporting Administrator's Guide

Document Release Date: June 2014
Software Release Date: June 2014



Legal Notices

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notice

© Copyright 1997 - 2014 Hewlett-Packard Development Company, L.P.

Trademark Notices

Adobe® is a trademark of Adobe Systems Incorporated.

Microsoft® and Windows® are U.S. registered trademarks of Microsoft Corporation.

UNIX® is a registered trademark of The Open Group.

Documentation Updates

The title page of this document contains the following identifying information:

- Software Version number, which indicates the software version.
- Document Release Date, which changes each time the document is updated.
- Software Release Date, which indicates the release date of this version of the software.

To check for recent updates or to verify that you are using the most recent edition of a document, go to: <http://h20230.www2.hp.com/selfsolve/manuals>

This site requires that you register for an HP Passport and sign in. To register for an HP Passport ID, go to: <http://h20229.www2.hp.com/passport-registration.html>

Or click the **New users - please register** link on the HP Passport login page.

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

Support

Visit the HP Software Support Online web site at: <http://www.hp.com/go/hpsoftwaresupport>

This web site provides contact information and details about the products, services, and support that HP Software offers.

HP Software online support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support web site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HP support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract. To register for an HP Passport ID, go to:

<http://h20229.www2.hp.com/passport-registration.html>

To find more information about access levels, go to:

http://h20230.www2.hp.com/new_access_levels.jsp

HP Software Solutions Now accesses the HPSW Solution and Integration Portal Web site. This site enables you to explore HP Product Solutions to meet your business needs, includes a full list of Integrations between HP Products, as well as a listing of ITIL Processes. The URL for this Web site is <http://h20230.www2.hp.com/sc/solutions/index.jsp>

Contents

- Chapter 1: Introduction 9
 - About Operational Reporting Content Pack 2.0 for PPM Center 9
 - About this Document 9
 - Operational Reporting Content on HP Live Network 10
 - Subscribing to Announcements on HPLN 11
 - Accessing Operational Reporting Documentation from HPLN 13
 - Related Documents 13
 - HP PPM Center Documents for PPM Center Content Pack 2.0 14

- Chapter 2: Deploying Operational Reporting on Windows Systems 15
 - Obtaining Installation Bundle for Operational Reporting Content Pack 2.0 15
 - Installing Operational Reporting on Windows Systems 15
 - Creating the Operational Reporting Database Schema 15
 - Loading PPM Center Data into the Operational Reporting Database 23

- Chapter 3: Deploying Operational Reporting on UNIX Systems 27
 - Obtaining Installation Bundle for Operational Reporting Content Pack 2.0 27
 - Installing Operational Reporting on UNIX Systems 27
 - Creating the Operational Reporting Database Schema 27
 - Loading PPM Center Data into the Operational Reporting Database 36

- Chapter 4: Upgrading Operational Reporting on Windows Systems 39
 - Preparing to Upgrade 39
 - Operational Reporting Upgrade for Windows Systems 40
 - Recovering from an Upgrade Failure 44
 - Upgrade Failure Resulting from Active PPM Servers 44
 - Failure During the Sample_preupgrade_rpt.bat Run 45
 - Failure During the Sample_upgrade_rpt.bat Run 45

- Chapter 5: Upgrading Operational Reporting on a UNIX System 46
 - Preparing to Upgrade 46
 - Operational Reporting Upgrade for UNIX Systems 47
 - Recovering from an Upgrade Failure 51

| | |
|--|-----------|
| Upgrade Failure Resulting from Active PPM Servers | 51 |
| Failure During the sample_preupgrade_rpt.sh Run | 52 |
| Failure During the sample_upgrade_rpt.sh Run | 52 |
| Chapter 6: Creating Views for Request Types and Table Components | 53 |
| Chapter 7: Refreshing Operational Reporting Data | 55 |
| Synchronizing Data in the Operational Reporting and PPM Center Database Schema | 55 |
| Running Incremental ETL Jobs | 55 |
| Checking ETL Job Progress | 55 |
| Verifying Successful Incremental ETL Jobs | 56 |
| Running Incremental ETL Jobs Manually | 56 |
| Purging Data | 58 |
| PPM Center Data Transfer During ETL | 58 |
| Common Dimension Data Transfer | 58 |
| HP Demand ManagementData Transfer | 58 |
| HP Portfolio Management Data Transfer | 58 |
| HP Project Management Data Transfer | 59 |
| HP Time Management Data Transfer | 59 |
| HP Resource Management Data Transfer | 59 |
| HP Financial Management Data Transfer | 59 |
| Date Range for Transferred Data | 59 |
| Date Range for Transferred HP Time Management Data | 60 |
| Date Range for Transferred HP Resource Management Data | 60 |
| Extending the Time Range of Resource Capacity Data | 61 |
| Recommendations for Running the Extend Data Script | 61 |
| Chapter 8: Operational Reporting Diagrams | 63 |
| Chapter 9: Definitions of Operational Reporting Tables | 70 |
| RPT_CALENDAR_DAYS | 70 |
| RPT_CM_FACT_CNTL | 71 |
| RPT_DEPLOYMENT | 71 |
| RPT_DIM_BUSINESS_OBJ_MV | 72 |
| RPT_DIM_FM_BENEFIT_MV | 73 |
| RPT_DIM_FM_BENEFIT_LINE_MV | 74 |
| RPT_DIM_FM_FCST_ACTUAL_LINE_MV | 75 |

RPT_DIM_FM_FCST_ACTUAL_MV78

RPT_DIM_FM_FIN_SUMMARY_MV81

RPT_DIM_FM_KNTA_LOOKUPS_MV83

RPT_DIM_ORG_UNITS_MV84

RPT_DIM_PM_DTL_CSTPARMS_IRS_MV87

RPT_DIM_PM_HDR_CSTPARMS_IRS_MV88

RPT_DIM_PM_PROJ_ISSUE_MV89

RPT_DIM_PM_PROJ_RISK_MV91

RPT_DIM_PM_PROJ_SCPCHG_MV93

RPT_DIM_PM_PROJ_SETTINGS_MV95

RPT_DIM_PM_TASKS96

RPT_DIM_PM_WORKPLAN_MV101

RPT_DIM_PORTFOLIOS102

RPT_DIM_PORTFOLIO_MANAGERS_MV103

RPT_DIM_PPM_FISCAL_PERIODS105

RPT_DIM_PROGRAM_CONTENT_MV106

RPT_DIM_PROGRAMS_MV107

RPT_DIM_PROJECTS_MV109

RPT_DIM_REGIONS_MV112

RPT_DIM_REQ_DTL_CUSTOM_PARAMS113

RPT_DIM_REQ_HDR_CUSTOM_PARAMS114

RPT_DIM_REQUESTS115

RPT_DIM_RM_POSITIONS119

RPT_DIM_RM_RESOURCEPOOLMGRS121

RPT_DIM_RM_RESOURCEPOOLS122

RPT_DIM_RM_RESOURCES125

RPT_DIM_RM_ROLES129

RPT_DIM_RM_STAFF_PROFILES130

RPT_DIM_TM_ACTIVITIES_MV132

RPT_DIM_TM_CHARGE_CODES_MV133

RPT_DIM_TM_PERIODS_MV134

RPT_DIM_TM_TIME_SHEET_LINES135

| | |
|--------------------------------------|-----|
| RPT_DIM_TM_TIME_SHEETS | 138 |
| RPT_DIM_WORKFLOW_STEPS | 139 |
| RPT_ETL_JOB | 140 |
| RPT_EVENT_DEF | 142 |
| RPT_EVENT_LOG | 142 |
| RPT_EVENT_LOG_DETAIL | 143 |
| RPT_FCT_FM_APPROVED_BUDGETS | 144 |
| RPT_FCT_FM_BENEFIT_CELLS | 146 |
| RPT_FCT_FM_FA_APPROVED_MV | 148 |
| RPT_FCT_FM_FCST_ACTUAL_CELL | 149 |
| RPT_FCT_PM_AGGR_PROJ_EFFORT | 151 |
| RPT_FCT_PM_AGGR_TASK_EFFORT | 152 |
| RPT_FCT_PM_TASK_ASSIGNMENTS | 154 |
| RPT_FCT_PM_TASK_COST | 156 |
| RPT_FCT_PM_TASK_UNASSGN_EFFORT | 159 |
| RPT_FCT_REQUEST_TRANSACTIONS | 161 |
| RPT_FCT_RM_RES_DISTRIBUTION | 162 |
| RPT_FCT_RM_RESOURCE_CAPACITY | 163 |
| RPT_FCT_RM_RESOURCE_DEMAND | 165 |
| RPT_FCT_RM_RESOURCE_DEMAND_STG | 168 |
| RPT_FCT_RM_RESOURCE_EFFORT | 170 |
| RPT_FCT_RM_RESRCE_CAPACITY_STG | 172 |
| RPT_FCT_RM_RSC_CAPACITY_AGGR | 174 |
| RPT_FCT_RM_RSC_DEMAND_AGGR | 176 |
| RPT_FCT_TM_ACTUAL_EFFORT | 179 |
| RPT_FCT_TM_COMPLIANCE | 181 |
| RPT_FCT_TM_COST | 184 |
| RPT_KNTA_USERS_DELETES | 186 |
| RPT_PARAMS | 186 |
| RPT_PM_PROJECT_ENTRIES | 187 |
| RPT_PM_PROJECT MANAGERS | 188 |
| RPT_PM_TASK_ENTRIES | 189 |

| | |
|--|------------|
| RPT_PPM_AGGR_WRK_DAYS_MV | 189 |
| RPT_RECREATE_INDEX | 190 |
| RPT_RERUN_STATUS | 191 |
| RPT_RM_FACT_CNTL | 192 |
| RPT_RM_RSC_RP_ENTRIES | 193 |
| RPT_RSC_RPDE_DELETES | 194 |
| RPT_TRACE_DETAILS | 195 |
| RPT_UPGRADE_PARAMS | 196 |
| Send Documentation Feedback | 198 |

Chapter 1: Introduction

IMPORTANT

Operational Reporting solution version Content Pack 2.0 is compatible with PPM Center versions 9.12~9.22. For compatibility matrix of Operational Reporting solution versions and PPM Center versions, see the latest Operational Reporting for PPM Center Release Notes.

- If you are an existing user of the Operational Reporting solution, see the corresponding upgrade section in this guide for instructions about how to upgrade your solution to version Content Pack 2.0.
- If you are deploying the Operational Reporting solution for the first time, make sure you have obtained the installation bundle for Content Pack 2.0, which contains everything from Operational Reporting solution version 9.10 to Content Pack 1.3 in addition to new features and enhancements implemented in Content Pack 2.0. For detailed installation instructions, see the corresponding deployment section in this guide.

BusinessObjects Universes and BusinessObjects reports that are delivered in Content Pack 1.3 are still working in Content Pack 2.0. You are free to customize these BusinessObjects Universes in order to use the new tables added for HP Demand Management and HP Portfolio Management in Content Pack 2.0.

About Operational Reporting Content Pack 2.0 for PPM Center

Operational Reporting Content Pack 2.0 for PPM Center is specific to Operational Reporting.

About this Document

This guide provides information about how to deploy the Operational Reporting solution for HP Project and Portfolio Management Center (PPM Center) and how to upgrade the Operational Reporting solution from earlier versions of the Operational Reporting solution to Content Pack 2.0. It is written for PPM Center administrators, configurators, and DBAs who are knowledgeable about PPM Center. Readers are assumed to be moderately knowledgeable about enterprise application development and skilled in enterprise system and database administration.

This chapter provides an overview of the components and structure of the Operational Reporting solution. The remaining chapters are as follows:

- ["Deploying Operational Reporting on Windows Systems" on page 15](#) provides the information you need to implement the Operational Reporting solution for PPM Center for the first time on a Windows system. It includes instructions for deploying Operational Reporting Content Pack 2.0 for PPM Center.
- ["Deploying Operational Reporting on UNIX Systems" on page 27](#) provides the information you need to implement the Operational Reporting solution for PPM Center for the first time on a UNIX system. It includes instructions for deploying Operational Reporting for PPM Center Content Pack 2.0.
- ["Upgrading Operational Reporting on Windows Systems" on page 39](#) provides instructions on how to upgrade your Operational Reporting deployment to Content Pack 2.0 on Windows systems. Operational Reporting Content Pack 1.3 is required before you can upgrade to Content Pack 2.0.
- ["Upgrading Operational Reporting on a UNIX System" on page 46](#) provides instructions on how to upgrade your Operational Reporting deployment to Content Pack 2.0 on UNIX systems. Operational Reporting Content Pack 1.3 is required before you can upgrade to Content Pack 2.0.
- ["Creating Views for Request Types and Table Components" on page 53](#) provides information about Operational Reporting creating views for request types and table components so that there is no mapping relationship between views and request types and table components.
- ["Refreshing Operational Reporting Data" on page 55](#), provides information about how to synchronize data in the PPM Center database schema and the Operational Reporting database schema.
- ["Operational Reporting Diagrams" on page 63](#) illustrates the relationship diagrams of Operational Reporting.
- ["Definitions of Operational Reporting Tables" on page 70](#) provides a detailed description of each table in the Operational Reporting schema. Refer to this chapter to obtain a detailed understanding of the underlying structure of Operational Reporting.

Operational Reporting Content on HP Live Network

HP Live Network (HPLN) is an online virtual community for product experts, partners, and customers to collaborate and share knowledge, best practices, and add-on content for HP software products, including PPM Center and Operational Reporting. You can log in to the Operational Reporting Community page on HPLN to access the latest news, updates, and documentation for Operational Reporting. You can browse from the Operational Reporting community page or subscribe to receive notifications via email.

Access to HPLN is free to all PPM Center customers. You must have an HP passport account to access the PPM Center and Operational Reporting community pages.

Note: Only project owners and administrators can post to the Announcements forum. If you are not a project owner or administrator, direct your feedback to the project owner or the general

discussion forum.

To access Operational Reporting content on HPLN:

1. Go to the [HP Support Contract information](http://support.openview.hp.com/entitlement/contracts) page (support.openview.hp.com/entitlement/contracts).
2. Sign in to the HP Passport page.

The HP Support Contract information page opens.

The screenshot shows the HP Support Contract information page. At the top left is the HP logo. The breadcrumb trail is 'Software > Software Support Online'. The main heading is 'Support contract information'. Below this is a sub-heading 'Support contract information' in a dark blue bar. The text below reads: 'Listed below are the software support contracts you have added to your HP Passport profile.' There is a table with columns: 'Contract', 'Expiration date', 'Status', and 'Delete'. Below the table is a section 'Add a contract to your profile' in a dark blue bar. This section contains a text input field labeled 'Contract identifier (SAID)*' and an 'Add »' button. Below this is a section 'Assistance options' with links: 'Investigate contract' and 'Contact HP sales'. On the left side, there is a sidebar with 'Contracts' (List contracts, Investigate contract, Contact HP sales, Help), 'Related links:' (Support home), and 'Site map'. At the bottom left of the sidebar is a 'Printable version' link.

Mana

3. In the **Contract identifier (SAID)*** box, type your service agreement ID (SAID).
4. Click **Add**.
5. Go to the [Operational Reports Content for Project and Portfolio Management](#) page on HP Live Network.
6. Log on to the HP Passport sign-in page.
7. To view the latest announcements about Operational Reporting, select the **Announcements** tab.

Subscribing to Announcements on HPLN

To subscribe to email notifications about new content on the Operational Reporting community page:

1. In the **Latest Announcements and Discussions** heading on the Operational Reporting community page, click the **Manage Notifications** link.


Operational Reports Content for Project and Portfolio Management

Welcome to the Operational Reporting Content delivery page. Operational Reporting for Project and Portfolio Management provides users with realistic examples of business reporting across the Project and Portfolio Management Center.


Overview

HP has developed Operational Reporting for Project and Portfolio Management based on SAP BusinessObjects. Note that an SAP BusinessObjects Enterprise XI 3.1 installation ships with the PPM Center media. For questions regarding this content, please see the Project and Portfolio Management discussion forums or open a support case, if needed.

Use the download link to get the PPM Center software installation bundle and documentation. HP recommends that you download the Operational Reporting Administrator's Guide and Operational Reporting Release Notes before you download the software.

 [Operational Report Download](#)

Latest Announcements and Discussions [\[Manage notifications \]](#) [\[View all \]](#)

[Operational reporting Content Announcements](#) 

- [Welcome!](#) [Mon, 20 Jun 2011 15:33:06 GMT]

Related Discussions

- [PPM Announcements and Discussions](#)

2. In the **Subscribed** column, select the check box for the Operational Reporting Content Announcements title.

Discussions > Manage notifications

Customers: For access, you must have an HP Passport account, and you must have entered in your products SAIDS, here: <http://support.openview.hp.com/entitlement/contracts>
HP Employees: For access, you must validate yourselves as an Employee, here: <http://support.openview.hp.com / Log in / edit profile / validate as employee>
Please refer to [the BSAEN and Lnc HPP Migration Guide](#) for further information, and the [Help and Support](#) pages.

| Subscribed | Title | Messages | Latest post | Notification |
|-------------------------------------|---|----------|-------------------------|--------------------|
| <input checked="" type="checkbox"/> | Operational reporting Content Announcements | 1 | 2011-06-21 15:00:05 GMT | Message-by-message |

Announcements for content updates for Operational Reporting content for PPM. These same announcements may also be sent to the main PPM Announcements forum along with others as desired by management. Announcements may be subscribed to via email or browsed via the web. Please note that announcement forums are one way - customers should use the relevant General Discussion forum as appropriate for any questions regarding these announcements.

Discussions per page: 25 Submit

Save changes Cancel

3. From the **Notification** list, select an option to indicate how you want to receive your notifications.
4. Click **Save Changes**.

Accessing Operational Reporting Documentation from HPLN

To access Operational Reporting documentation from HPLN:

1. Under **Quick Links**, click **Download Reporting Content**.
2. In the **Name** column, click the link for the document you want to download.

The Operational Reporting community page also provides links to pages where you can submit a support ticket, access HP Support Online, and search the support knowledge base.

Related Documents

This section lists HP documents that contain useful information for Operational Reporting administrators and users.

HP PPM Center Documents for PPM Center Content Pack 2.0

- *Release Notes*
- *Deployment Best Practices for Operational Reporting*
- *Operational Reporting User's Guide*
- *System Requirements and Compatibility Matrix*

Before you start to deploy Operational Reporting, check the *System Requirements and Compatibility Matrix* document to make sure that your operating environment meets *all* of the minimum system requirements.

- *Data Model Guide*

The *Data Model Guide* provides details about the internal structure of the data models for both PPM Center and Operational Reporting.

To obtain any of the HP PPM Center documents listed, go to the Software Product Manuals Web site (support.openview.hp.com/selfsolve/manuals). To access this Web site, you must first set up an PPM Center Passport account.

Chapter 2: Deploying Operational Reporting on Windows Systems

Obtaining Installation Bundle for Operational Reporting Content Pack 2.0

To obtain the installation bundle for Operational Reporting Content Pack 2.0 for PPM Center,

1. Go to [Operational Reports Content for Project and Portfolio Management - Downloads](http://hp1n.hp.com/node/81/contentfiles) page on HPLN Web site ([//hp1n.hp.com/node/81/contentfiles](http://hp1n.hp.com/node/81/contentfiles)).

Note: To access this Web site, you must provide your SAID for PPM Center.

2. Select **PPM Operational Reporting CP2.0**.
3. Download the **CP2.0_Refresh_Install_Bundle.zip** file.

Extract the entire contents of **CP2.0_Refresh_Install_Bundle.zip** to the *<Op_Reports_Home>* folder.

Installing Operational Reporting on Windows Systems

- ["Creating the Operational Reporting Database Schema" below](#)
- ["Loading PPM Center Data into the Operational Reporting Database" on page 23](#)

Creating the Operational Reporting Database Schema

To create the Operational Reporting database schema, you run the setup script.

1. Stop all PPM Servers (including all nodes in a server cluster).

Caution: If the `REMOTE_ADMIN_REQUIRE_AUTH` parameter is set to `true`, users running `kStop.bat` to shut down the PPM Server must supply a valid PPM Center user name and

password. If the parameter is set to `false`, any user with access to the `kStop.bat` script can shut down the server. For information about the `REMOTE_ADMIN_REQUIRE_AUTH` parameter, see the *Installation and Administration Guide*.

To stop a PPM Server:

- a. From the Control Panel, select **Administrative Tools > Services**.
- b. In the Services window, right-click the HP PPM service, and then click **Stop** on the shortcut menu.

Note: In the Windows services window, the service name begins with "HP PPM".

2. Grant necessary privileges to PPM Schema by one of the following methods:

- Connect to PPM database as SYSDBA and run the SQL commands as follows:

```
grant select_catalog_role to <PPM_SCHEMA>;  
grant execute_catalog_role to <PPM_SCHEMA>;  
grant create_job to <PPM_SCHEMA>;  
grant create_materialized_view to <PPM_SCHEMA>;  
grant dba to <PPM_SCHEMA>;  
grant become_user to <PPM_SCHEMA>;  
grant create_view to <PPM_SCHEMA>;  
grant create_sequence to <PPM_SCHEMA>;
```

- Run the `sample_setup_ppm_sys.bat` script:

- i. Log on to the machine that can connect to PPM Center database and Reporting database, navigate to the `<Op_Report_Home>\install\sample` directory, and open the `sample_setup_ppm_sys.bat` file in a text editor.
- ii. Uncomment the parameters listed in the following table, replace the placeholders with valid values, and then save and close the file.

| Parameter | Value |
|-------------------------|---|
| SYS user name of PPM DB | SYS user name for the PPM Center database Example value: sys |

| Parameter | Value |
|--|---|
| PPM DB Schema Name | <p>This value should exist in the Oracle <code>tnsnames.ora</code> entry.</p> <p>Example value: PPM_SCHEMA</p> <p>Important: The PPM Center database schema name must be in all capital letters.</p> |
| Full <code>tnsnames.ora</code> entry to PPM schema | <p>Full <code>tnsnames.ora</code> entry for the PPM Center database schema</p> <ul style="list-style-type: none"> For <code>HOST</code>, specify the IP address of the PPM Center database server For <code>PORT</code>, specify the PPM Center database port For <code>SERVICE_NAME</code>, specify the SID in <code>tnsnames.ora</code> file for the PPM Center database <p>Example value:</p> <pre>"(DESCRIPTION= (ADDRESS= (PROTOCOL=TCP) (HOST=16.89.27.63) (PORT=1522)) (CONNECT_DATA= (SERVER=dedicated) (SERVICE_NAME=MDB1106A)))"</pre> |

- iii. Run the `sample_setup_ppm_sys.bat` script.
- iv. During the script run, provide the following information when prompted:
 PPM Center database server SYS user password
- v. Navigate to the `<Op_Report_Home>\install\log` directory and check the `setup_ppm_sys.log` file for errors.

3. Create an empty reporting schema and grant necessary privileges to it by one of the following methods:

- Connect to Report database as SYSDBA and run the SQL commands as follows:
 - i. Create a new schema:

```
CREATE USER <report_schema_name>
IDENTIFIED BY <report_shcema_password>
DEFAULT TABLESPACE <data_table_space>
TEMPORARY TABLESPACE <temp_table_space>
```

```
QUOTA UNLIMITED ON <data_table_space>  
QUOTA UNLIMITED ON <index_table_space>  
QUOTA UNLIMITED ON <DATA_NOLOGGING_TABLESPACE>  
QUOTA UNLIMITED ON <INDEX_NOLOGGING_TABLESPACE>;
```

- ii. Grant necessary privileges to the new schema:

```
grant connect to <reporting_shcema_name>;  
grant create procedure to <reporting_shcema_name>;  
grant create session to <reporting_shcema_name>;  
grant create sequence to <reporting_shcema_name>;  
grant create synonym to <reporting_shcema_name>;  
grant create table to <reporting_shcema_name>;  
grant create view to <reporting_shcema_name>;  
grant create materialized view to <reporting_shcema_name>;  
grant create database link to <reporting_shcema_name>;  
grant alter session to <reporting_shcema_name>;  
grant analyze any to <reporting_shcema_name>;  
grant select on v_$parameter to <reporting_shcema_name>;  
grant create job to <reporting_shcema_name>;  
grant EXECUTE ANY PROGRAM to <reporting_shcema_name>;  
grant MANAGE SCHEDULER to <reporting_shcema_name>;  
grant select on dba_scheduler_programs to <reporting_shcema_name>;  
grant select on dba_scheduler_schedules to <reporting_shcema_name>;  
grant select on dba_scheduler_jobs to <reporting_shcema_name>;  
DBMS_RULE_ADM.GRANT_SYSTEM_PRIVILEGE(DBMS_RULE_ADM.CREATE_RULE_OBJ,  
'<reporting_shcema_name>');  
DBMS_RULE_ADM.GRANT_SYSTEM_PRIVILEGE (DBMS_RULE_ADM.CREATE_RULE_SET_  
OBJ,  
'<reporting_shcema_name>');  
DBMS_RULE_ADM.GRANT_SYSTEM_PRIVILEGE (DBMS_RULE_ADM.CREATE_  
EVALUATION_CONTEXT_  
OBJ, '<reporting_shcema_name>');
```

- o Run the `sample_setup_reporting_sys.bat` script:
 - A. Navigate to the `<Op_Report_Home>\install\sample` directory, and open the `sample_setup_reporting_sys.bat` file in a text editor.
 - B. Uncomment the parameters listed in the following table, replace the placeholders with valid values, and then save and close the file.

| Parameter | Value |
|-------------------------------------|--|
| SYS user name of Reporting DB | SYS user name for the Operational Reporting database Example value: sys |
| Reporting DB Schema Name | Operational Reporting database schema name Example value: RPT_SCHEMA Important: The Operational Reporting database schema name must be in all capital letters. |
| Reporting DB data tablespace name | Operational Reporting database to store data. Example value: RPT_DATA_TS Important: The Operational Reporting database data tablespace name must be in all capital letters. |
| Reporting DB TEMP tablespace name | Operational Reporting database temp tablespace Example value: RPT_TEMP_TS Important: The Operational Reporting database temp tablespace name must be in all capital letters. |
| Reporting DB INDEX tablespace name | Operational Reporting database to store index Example value: RPT_INDEX_TS Important: The Operational Reporting database index tablespace name must be in all capital letters. |
| Reporting DB TNS Name | Identifies the Oracle instance that runs the Operational Reporting database schema. TNS name is configured in the tnsnames.ora file. Example value: RPT |
| RPT_DATA_NOLOGGING_TABLESPACE_NAME | Separate tablespace that requires no redo log for the Operational Reporting database to store data. Example value: RPT_DATA_TS_NL |
| RPT_INDEX_NOLOGGING_TABLESPACE_NAME | Separate tablespace that requires no redo log for the Operational Reporting database to store indexes. Example value: RPT_INDEX_TS_NL |

- C. Run the `sample_setup_reporting_sys.bat` script.
- D. During the script run, provide the following information when prompted:
 PPM Center database server SYS user password
- E. Navigate to the `<Op_Report_Home>\install\log` directory and check the `setup_reporting_sys.log` file for errors.

Note: Running the `sample_setup_reporting_sys.bat` script drops the old reporting schema and creates a new reporting schema. The new reporting schema is reusable, therefore, you do not need to run this script if you want to re-install the Operational Reporting package.

- 4. Navigate to the `<Op_Report_Home>\install\sample` directory, and open the `sample_setup_all.bat` file in a text editor.
- 5. Uncomment the parameters listed in the following table, replace the placeholders with valid values, and then save and close the file.

| Information | Description |
|-----------------------------------|---|
| Reporting DB Schema Name | Operational Reporting database schema name Example value: RPT_SCHEMA Important: The Operational Reporting database schema name must be in all capital letters. |
| Reporting DB TNS Name | Identifies the Oracle instance that runs the Operational Reporting database schema. TNS name is configured in the <code>tnsnames.ora</code> file. Example value: RPT |
| Reporting DB data_tableSPACE_name | Name of the data tablespace for the Operational Reporting database Example value: RPT_DATA_TS Important: The Operational Reporting database data tablespace name must be in all capital letters. |

| Information | Description |
|--|--|
| Reporting DB temp_ tablespace_name | <p>Name of the temp tablespace for the Operational Reporting database</p> <p>Example value: RPT_TEMP_TS</p> <p>Important: The Operational Reporting database temp tablespace name must be in all capital letters.</p> |
| Reporting DB index_ tablespace_name | <p>Name of the index tablespace for the Operational Reporting database</p> <p>Example value: RPT_INDEX_TS</p> <p>Important: The Operational Reporting database index tablespace name must be in all capital letters.</p> |
| PPM DB Schema Name | <p>PPM Center database schema name.</p> <p>This value should exist in the Oracle <code>tnsnames.ora</code> entry.</p> <p>Example value: PPM_SCHEMA</p> <p>Important: The PPM Center database schema name must be in all capital letters. If the name contains any lowercase characters, an error occurs.</p> |
| PPM DB data_ tablespace_name | <p>PPM Center database data tablespace name.</p> <p>Note: This refers to the existing data tablespace in the <i>PPM Center database schema</i>. The PPM Center schema stores this in the <code>KINS_TABLESPACES</code> table.</p> <p>Example value: PPM_DATA_TS</p> <p>Important: The PPM Center database data tablespace name must be in all capital letters.</p> |

| Information | Description |
|---------------------------------------|--|
| PPM DB temp_ tablespace_name | <p>PPM Center database temp tablespace name</p> <p>Note: This refers to the existing temp tablespace in the <i>PPM Center database schema</i>. The PPM Center schema stores this in the KINS_TABLESPACES table.</p> <p>Example value: PPM_TEMP_TS</p> <p>Important: The PPM Center database temp tablespace name must be in all capital letters.</p> |
| PPM DB index_ tablespace_name | <p>PPM Center database index tablespace name</p> <p>Note: This refers to the existing index tablespace in the <i>PPM Center database schema</i>. The PPM Center schema stores this in the KINS_TABLESPACES table.</p> <p>Example value: PPM_INDEX_TS</p> <p>Important: The PPM Center database index tablespace name must be in all capital letters.</p> |
| Full tnsnames.ora entry to PPM schema | <p>Full tnsnames.ora entry for the PPM Center database schema</p> <ul style="list-style-type: none"> ■ For HOST, specify the IP address of the PPM Center database server ■ For PORT, specify the PPM Center database port ■ For SERVICE_NAME, specify the SID in tnsnames.ora file for the PPM Center database <p>Example value:</p> <pre>"(DESCRIPTION= (ADDRESS= (PROTOCOL=TCP) (HOST=16.89.27.63) (PORT=1522)) (CONNECT_DATA= (SERVER=dedicated) (SERVICE_NAME=MDB1106A)))"</pre> |
| DB_LINK_NAME to PPM | <p>Name of the link to the PPM Center database</p> <p>This value is generated in the Operational Reporting database schema.</p> <p>Example value: PPM_DB_LINK</p> <p>Important: The name of the link to the PPM Center database must be in all capital letters.</p> |

| Information | Description |
|---|--|
| Reporting DB DATA_ NOLOGGING_ TABLESPACE | Separate tablespace that requires no redo log for the Operational Reporting database to store data. Example value: RPT_DATA_TS_NL |
| Reporting DB INDEX_ NOLOGGING_ TABLESPACE | Separate tablespace that requires no redo log for the Operational Reporting database to store indexes. Example value: RPT_INDEX_TS_NL |

6. Run the `sample_setup_all.bat` script.
7. During the script run, provide the following information when prompted:
 - PPM Center database server schema password
 - Operational Reporting database server schema password
8. The script performs a sanity check on PPM database. Do one of the following:
 - If the sanity check fails, an error message pops up. HP strongly recommends that you fix the errors by the suggestions on the error message. Then, continue with [Step 9](#).
 - If the sanity check passes, continue with [Step 9](#).
9. Navigate to the `<Op_Report_Home>\install\log` directory and check the `setup_all.log` file for errors. If the `setup_all.log` file indicates that compilation errors occurred, run the following:


```
Select * from user_objects where status = 'INVALID'
```

If no rows are returned, you can safely ignore the warning.
10. Revoke system privileges from PPM Schema.

Connect to PPM database as SYSDBA and run the SQL commands as follows:

```
revoke dba from <PPM_SCHEMA>;
revoke become user from <PPM_SCHEMA>;
```
11. Restart the PPM Servers.

Loading PPM Center Data into the Operational Reporting Database

After you create the Operational database schema, and synchronize the tables and data, you can import your PPM Center data into the Operational Reporting database. This section provides

information about how to run the load script that brings PPM Center data into the Operational Reporting database schema.

To run the load script:

1. Gather the information listed in the following table.

| Parameter | Description |
|-------------------------------------|--|
| Reporting DB Schema Name | Operational Reporting database schema name Example value: RPT_SCHEMA Important: The Operational Reporting database schema name must be in all capital letters. |
| Reporting DB TNS Name | Identifies the Oracle instance running the Operational Reporting database schema. The TNS name is configured in the <code>tnsnames.ora</code> file. Example value: RPT |
| Reporting DB index_ tablespace_name | Name of the index tablespace for the Operational Reporting database Example value: RPT_INDEX_TS Important: The Operational Reporting database index tablespace name must be in all capital letters. |
| DB_LINK_NAME to PPM | Name of the link to the PPM Center database. This link is created automatically during the <code>setup_all</code> script run. Example value: PPM_DB_LINK Important: The name of the link to the PPM Center database must be in all capital letters. |
| ETL start date (mm-dd-yyyy) | Start date (in mm-dd-yyyy format) for the PPM Center data to load into the Operational Reporting database schema. Example value: 01-01-2010 |
| ETL end date (mm-dd-yyyy) | End date (in mm-dd-yyyy format) for the PPM Center data to load into the Operational Reporting database schema. Note: The ETL end date you specify is converted based on the fiscal year. For details, see the <i>Installation and Administration Guide</i> . Example value: 01-01-2011 |

| Parameter | Description |
|---|--|
| Reporting DB data_ tablespace_name | Name of the data tablespace for the Operational Reporting database Example value: RPT_DATA_TS Important: The name of the link to the PPM Center database must be in all capital letters. |
| Request dimension ETL start date (mm-dd-yyyy) | Start date (in mm-dd-yyyy format) for the PPM Center request data to load into the Operational Reporting database schema. Example value: 01-01-2010 Note: If your PPM Center database contains data for old, but active requests, you can include that data without importing all data from that time period. |
| Reporting DB DATA_ NOLOGGING_ TABLESPACE | Separate tablespace that requires no redo log for the Operational Reporting database to store data. Example value: RPT_DATA_TS_NL |
| Reporting DB INDEX_ NOLOGGING_ TABLESPACE | Separate tablespace that requires no redo log for the Operational Reporting database to store indexes. Example value: RPT_INDEX_TS_NL |

2. Navigate to the <Op_Report_Home>\install\sample directory, and open the sample_load_data.bat file in a text editor.
3. Replace each of the variables in the load script with the corresponding values you prepared in [Step 1](#) and then save and close the file.
4. Navigate to the <Op_Report_Home>\install\sample directory, and run the sample_load_data.bat script.
5. During the load script run, provide Operational Reporting database schema password and the Operational Reporting SYS user password, as prompted.
6. The script creates a load_data.log file in the <Op_Report_Home>\install\log directory. Check the log file to make sure that no errors occurred.

Note: You can run the sample_load_data.bat only once. If you discover any incorrect data, you have to re-install the Operational Reporting package. To re-install the package, you start from [Step 4](#) in "Creating the Operational Reporting Database Schema" on page 15 and then run the load script.

Chapter 3: Deploying Operational Reporting on UNIX Systems

Obtaining Installation Bundle for Operational Reporting Content Pack 2.0

To obtain the installation bundle for Operational Reporting Content Pack 2.0 for PPM Center,

1. Go to [Operational Reports Content for Project and Portfolio Management - Downloads](http://hp1n.hp.com/node/81/content-files) page on HPLN Web site (hp1n.hp.com/node/81/content-files).

Note: To access this Web site, you must provide your SAID for PPM Center.

2. Select **PPM Operational Reporting CP2.0**.
3. Download the **CP2.0_Refresh_Install_Bundle.tar.gz** file.

Extract the entire contents of **CP2.0_Refresh_Install_Bundle.tar.gz** to the *<Op_Reports_Home>* folder.

Installing Operational Reporting on UNIX Systems

- ["Creating the Operational Reporting Database Schema" below](#)
- ["Loading PPM Center Data into the Operational Reporting Database" on page 36](#)

Creating the Operational Reporting Database Schema

To create the Operational Reporting database schema, you run the setup script.

1. Stop all PPM Servers (including all nodes in a server cluster).

Caution: If the `REMOTE_ADMIN_REQUIRE_AUTH` parameter is set to `true`, users running

`kStop.sh` to shut down the PPM Server must supply a valid PPM Center user name and password. If the parameter is set to `false`, any user with access to the `kStop.sh` script can shut down the server. For information about the `REMOTE_ADMIN_REQUIRE_AUTH` parameter, see the *Installation and Administration Guide*.

To stop a PPM Server:

- a. Navigate to the `<PPM_Home>/bin` directory.
- b. Run the `kStop.sh` script as follows:

```
sh ./kStop.sh -now -user <User_Name>
```

Make sure that you type a valid user name for a user who has Administrator privileges.

2. Grant necessary privileges to PPM Schema by one of the following methods:

- Connect to PPM database as SYSDBA and run the SQL commands as follows:

```
grant select_catalog_role to <PPM_SCHEMA>;  
grant execute_catalog_role to <PPM_SCHEMA>;  
grant create_job to <PPM_SCHEMA>;  
grant create_materialized_view to <PPM_SCHEMA>;  
grant dba to <PPM_SCHEMA>;  
grant become_user to <PPM_SCHEMA>;  
grant create_view to <PPM_SCHEMA>;  
grant create_sequence to <PPM_SCHEMA>;
```

- Run the `sample_setup_ppm_sys.sh` script:

- i. Log on to the machine that can connect to PPM Center database and Reporting database, navigate to the `<Op_Report_Home>/install/sample` directory, and open the `sample_setup_ppm_sys.sh` file in a text editor.
- ii. Uncomment the parameters listed in the following table, replace the placeholders with valid values, and then save and close the file.

| Parameter | Value |
|-------------------------|---|
| SYS user name of PPM DB | SYS user name for the PPM Center database Example value: sys |

| Parameter | Value |
|--|---|
| PPM DB Schema Name | <p>This value should exist in the Oracle <code>tnsnames.ora</code> entry.</p> <p>Example value: PPM_SCHEMA</p> <p>Important: The PPM Center database schema name must be in all capital letters.</p> |
| Full <code>tnsnames.ora</code> entry to PPM schema | <p>Full <code>tnsnames.ora</code> entry for the PPM Center database schema</p> <ul style="list-style-type: none"> • For <code>HOST</code>, specify the IP address of the PPM Center database server • For <code>PORT</code>, specify the PPM Center database port • For <code>SERVICE_NAME</code>, specify the SID in <code>tnsnames.ora</code> file for the PPM Center database <p>Example value:</p> <pre>"(DESCRIPTION= (ADDRESS= (PROTOCOL=TCP) (HOST=16.89.27.63) (PORT=1522)) (CONNECT_DATA= (SERVER=dedicated) (SERVICE_NAME=MDB1106A)))"</pre> |

- iii. Run the `sample_setup_ppm_sys.sh` script.
- iv. During the script run, provide the following information when prompted:

PPM Center database server SYS user password
- v. Navigate to the `<Op_Report_Home>/install/log` directory and check the `setup_ppm_sys.log` file for errors.

3. Create an empty reporting schema and grant necessary privileges to it by one of the following methods:

- Connect to Report database as SYSDBA and run the SQL commands as follows:

- i. Create a new schema:

```
CREATE USER <report_schema_name>
IDENTIFIED BY <report_schema_password>
DEFAULT TABLESPACE <data_table_space>
TEMPORARY TABLESPACE <temp_table_space>
QUOTA UNLIMITED ON <data_table_space>
```

```
QUOTA UNLIMITED ON <index_table_space>  
QUOTA UNLIMITED ON <DATA_NOLOGGING_TABLESPACE>  
QUOTA UNLIMITED ON <INDEX_NOLOGGING_TABLESPACE>;
```

ii. Grant necessary privileges to the new schema:

```
grant connect to <reporting_shcema_name>;  
grant create procedure to <reporting_shcema_name>;  
grant create session to <reporting_shcema_name>;  
grant create sequence to <reporting_shcema_name>;  
grant create synonym to <reporting_shcema_name>;  
grant create table to <reporting_shcema_name>;  
grant create view to <reporting_shcema_name>;  
grant create materialized view to <reporting_shcema_name>;  
grant create database link to <reporting_shcema_name>;  
grant alter session to <reporting_shcema_name>;  
grant analyze any to <reporting_shcema_name>;  
  
grant select on v_$parameter to <reporting_shcema_name>;  
grant create job to <reporting_shcema_name>;  
grant EXECUTE ANY PROGRAM to <reporting_shcema_name>;  
grant MANAGE SCHEDULER to <reporting_shcema_name>;  
grant select on dba_scheduler_programs to <reporting_shcema_name>;  
grant select on dba_scheduler_schedules to <reporting_shcema_name>;  
grant select on dba_scheduler_jobs to <reporting_shcema_name>;  
  
DBMS_RULE_ADM.GRANT_SYSTEM_PRIVILEGE(DBMS_RULE_ADM.CREATE_RULE_OBJ,  
'<reporting_shcema_name>');  
DBMS_RULE_ADM.GRANT_SYSTEM_PRIVILEGE (DBMS_RULE_ADM.CREATE_RULE_SET_OBJ,  
'<reporting_shcema_name>');  
DBMS_RULE_ADM.GRANT_SYSTEM_PRIVILEGE (DBMS_RULE_ADM.CREATE_EVALUATION_  
CONTEXT_  
OBJ, '<reporting_shcema_name>');
```

■ Run the `sample_setup_reporting_sys.sh` script:

- i. Navigate to the `<Op_Report_Home>\install\sample` directory, and open the `sample_setup_reporting_sys.sh` file in a text editor.
- ii. Uncomment the parameters listed in the following table, replace the placeholders with valid values, and then save and close the file.

| Parameter | Value |
|--|--|
| SYS user name of Reporting DB | SYS user name for the Operational Reporting database Example value: sys |
| Reporting DB Schema Name | Operational Reporting database schema name Example value: RPT_SCHEMA Important: The PPM Center database schema name must be in all capital letters. |
| Reporting DB data_tablespace_name | Operational Reporting database to store data. Example value: RPT_DATA_TS Important: The Operational Reporting database tablespace name must be in all capital letters. |
| Reporting DB temp_tablespace_name | Operational Reporting database temp tablespace Example value: RPT_TEMP_TS Important: The Operational Reporting database temp tablespace name must be in all capital letters. |
| Reporting DB index_tablespace_name | Operational Reporting database to store index Example value: RPT_INDEX_TS Important: The Operational Reporting database index tablespace name must be in all capital letters. |
| Reporting DB TNS Name | Identifies the Oracle instance that runs the Operational Reporting database schema. TNS name is configured in the tnsnames.ora file. Example value: RPT |
| Reporting DB DATA_NOLOGGING_TABLESPACE | Separate tablespace that requires no redo log for the Operational Reporting database to store data. Example value: RPT_DATA_TS_NL |

| Parameter | Value |
|---|--|
| Reporting DB INDEX_NOLOGGING_TABLESPACE | Separate tablespace that requires no redo log for the Operational Reporting database to store indexes. Example value: RPT_INDEX_TS_NL |

- iii. Run the `sample_setup_reporting_sys.sh` script.
- iv. During the script run, provide the following information when prompted:

PPM Center database server SYS user password
- v. Navigate to the `<Op_Report_Home>\install\log` directory and check the `setup_reporting_sys.log` file for errors.

Note: Running the `sample_setup_reporting_sys.bat` script drops the old reporting schema and creates a new reporting schema. The new reporting schema is reusable, therefore, you do not need to run this script if you want to re-install the Operational Reporting package.

- 4. Navigate to the `<Op_Report_Home>/install/sample` directory, and open the `sample_setup_all.sh` file in a text editor.

Note: Make sure that you make the file as an executable.
 For example: `chmod +x sample_setup_all.sh`

- 5. Uncomment the parameters listed in the following table, replace the placeholders with valid values, and then save and close the file.

| Parameter | Description |
|--------------------------|---|
| Reporting DB Schema Name | Operational Reporting database schema name Example value: RPT_SCHEMA Important: The Operational Reporting database schema name must be in all capital letters. |

| Parameter | Description |
|------------------------------------|---|
| Reporting DB data_tablespace_name | Name of the data tablespace for the Operational Reporting database Example value: RPT_DATA_TS Important: The Operational Reporting database data tablespace name must be in all capital letters. |
| Reporting DB temp_tablespace_name | Name of the temp tablespace for the Operational Reporting database Example value: RPT_TEMP_TS Important: The Operational Reporting database temp tablespace name must be in all capital letters. |
| Reporting DB index_tablespace_name | Name of the index tablespace for the Operational Reporting database Example value: RPT_INDEX_TS Important: The Operational Reporting database index tablespace name must be in all capital letters. |
| Reporting DB TNS Name | Identifies the Oracle instance that runs the Operational Reporting database schema. TNS name is configured in the tnsnames.ora file. Example value: RPT |
| PPM DB Schema Name | PPM Center database schema name. This value should exist in the Oracle tnsnames.ora entry. Example value: PPM_SCHEMA Important: The PPM Center database schema name must be in all capital letters. |

| Parameter | Description |
|------------------------------|--|
| PPM DB data_tablespace_name | PPM Center database data tablespace name Note: This refers to the existing data tablespace in the <i>PPM Center database schema</i> . The PPM Center schema stores this in KINS_TABLESPACES table. Example value: PPM_DATA_TS Important: The PPM Center database data tablespace name must be in all capital letters. |
| PPM DB temp_tablespace_name | PPM Center database temp tablespace name Note: This refers to the existing temp tablespace in the <i>PPM Center database schema</i> . The PPM Center schema stores this in KINS_TABLESPACES table. Example value: PPM_TEMP_TS Important: The PPM Center database temp tablespace name must be in all capital letters. |
| PPM DB index_tablespace_name | PPM Center database index tablespace name Note: This refers to the existing index tablespace in the <i>PPM Center database schema</i> . The PPM Center schema stores this in KINS_TABLESPACES table. Example value: PPM_INDEX_TS Important: The PPM Center database index tablespace name must be in all capital letters. |

| Parameter | Description |
|---|--|
| Full tnsnames.ora entry to PPM schema | <p>Full tnsnames.ora entry for the PPM Center database schema</p> <ul style="list-style-type: none"> ■ For HOST, specify the IP address of the PPM Center database server ■ For PORT, specify the PPM Center database port ■ For SERVICE_NAME, specify the SID in tnsnames.ora file for the PPM Center database <p>Example value:</p> <pre>"(DESCRIPTION= (ADDRESS= (PROTOCOL=TCP) (HOST=16.89.27.63) (PORT=1522)) (CONNECT_DATA= (SERVER=dedicated) (SERVICE_NAME=MDB1106A)))"</pre> |
| DB_LINK_NAME to PPM | <p>Name of the link to the PPM Center database</p> <p>This value is generated in the Operational Reporting database schema.</p> <p>Example value: PPM_DB_LINK</p> <p>Important: The name of the link to the PPM Center database must be in all capital letters.</p> |
| Reporting DB DATA_NOLOGGING_TABLESPACE | <p>Separate tablespace that requires no redo log for the Operational Reporting database to store data. Example value: PPM_DATA_TS_NL</p> |
| Reporting DB INDEX_NOLOGGING_TABLESPACE | <p>Separate tablespace that requires no redo log for the Operational Reporting database to store indexes.</p> <p>Example value: RPT_INDEX_TS_NL</p> |

6. Run the sample_setup_all.sh script.
7. During the script run, provide the following passwords when prompted:
 - PPM Center database server schema password
 - Operational Reporting database server schema password
8. The script performs a sanity check on PPM database. Do one of the following:

- If the sanity check fails, an error message pops up. You must fix the errors by the suggestions on the error message.
 - If the sanity check passes, continue with [Step 9](#).
9. The script run creates a log file in the `<Op_Report_Home>/install/log` directory and check the `setup_all.log` file to make sure that no errors occurred. If the `setup_all.log` file indicates that compilation errors occurred, run the following:

```
Select * from user_objects where status = 'INVALID'
```

If no rows are returned, you can safely ignore the warning.

10. Revoke system privileges from PPM Schema.

Connect to PPM database as SYSDBA and run the SQL commands as follows:

```
revoke dba from <PPM_SCHEMA>;  
revoke become user from <PPM_SCHEMA>;
```

11. Restart the PPM Servers.

Caution: If your PPM Center instance includes multiple nodes in a cluster configuration, allow 10 to 15 seconds between node start-ups. For information about how to start PPM Servers, see the *Installation and Administration Guide*.

Loading PPM Center Data into the Operational Reporting Database

After you create the Operational database schema, and synchronized the tables and data, you can import your PPM Center data into the Operational Reporting database. This section provides information about how to run the load script that brings PPM Center data into the Operational Reporting database schema

To run the load script:

1. Gather the information listed in the following table.

| Parameter | Description |
|------------------------------------|--|
| Reporting DB Schema Name | Operational Reporting database schema name Example value: RPT_SCHEMA |
| Reporting DB TNS Name | Identifies the Oracle instance running the Operational Reporting database schema. the TNS name is configured in the <code>tnsnames.ora</code> file. Example value: RPT |
| Reporting DB index_tablespace_name | Name of the index tablespace for the Operational Reporting database Example value: RPT_INDEX_TS Important: The Operational Reporting database index tablespace name must be in all capital letters. |
| DB_LINK_NAME to PPM | Name of the link to the PPM Center database. This link is created automatically during the <code>setup_all</code> script run. Example value: PPM_DB_LINK |
| ETL start date (mm-dd-yyyy) | Start date (in mm-dd-yyyy format) for the PPM Center data to load into the Operational Reporting database schema. Example value: 01/01/2009 |
| ETL end date (mm-dd-yyyy) | End date (in mm-dd-yyyy format) for the PPM Center data to load into the Operational Reporting database schema. Example value: 01/01/2011 Note: The ETL end date you specify is converted based on the fiscal year. For details, see the <i>Installation and Administration Guide</i> . |
| Reporting DB data_tablespace_name | Name of the data tablespace for the Operational Reporting database Example value: RPT_DATA_TS Important: The Operational Reporting database data tablespace name must be in all capital letters. |

| Parameter | Description |
|---|--|
| Request dimension ETL start date (mm-dd-yyyy) | Start date (in mm-dd-yyyy format) for the PPM Center request data to load into the Operational Reporting database schema. Example value: 01/01/2009 Note: If your PPM Center database contains data for old, but active requests, you can include that data without importing all data from that time period. |
| Reporting DB DATA_NOLOGGING_TABLESPACE | Separate tablespace that requires no redo log for the Operational Reporting database to store data. Example value: PPM_DATA_TS_NL |
| Reporting DB INDEX_NOLOGGING_TABLESPACE | Separate tablespace that requires no redo log for the Operational Reporting database to store indexes. Example value: PPM_INDEX_TS_NL |

2. Log on to the machine that can connect to PPM Center database and Reporting database, navigate to the `<Op_Report_Home>/install/sample` directory, and open the `sample_load_data.sh` file in a text editor.

Note: Make sure that you make the file executable.
For example: `chmod +x sample_load_data.sh`

3. Replace each of the variables in the load script with the corresponding values you prepared for [Step 1](#), and then save and close the file.
4. Navigate to the `<Op_Report_Home>/install/sample` directory, and run `sample_load_data.sh` script.
5. During the load script run, provide Operational Reporting database schema password and the Operational Reporting SYS user password, as prompted.
6. The script creates a `load_data.log` file in the `<Op_Report_Home>/install/log` directory. Check the log file to make sure that no errors occurred.

Note: You can run the `sample_load_data.bat` only once. If you discover any incorrect data, you have to re-install the Operational Reporting package. To re-install the package, you start from [Step 4](#) in "Creating the Operational Reporting Database Schema" on [page 27](#) and then run the load script.

Chapter 4: Upgrading Operational Reporting on Windows Systems

Preparing to Upgrade

This section addresses the tasks to complete before you begin to upgrade to PPM Center version Content Pack 2.0.

As an existing user of the Operational Reporting solution, you can upgrade the Operational Reporting only from Content Pack 1.3 to Content Pack 2.0. For instructions about upgrading Operational Reporting solution to Content Pack 1.3, see *Operational Reporting Administrator's Guide* for Content Pack 1.3.

Note: If you are deploying the Operational Reporting solution for the first time, follow the instructions for deployment provided in "[Deploying Operational Reporting on Windows Systems](#)" on page 15.

To prepare to upgrade Operational Reporting:

1. Log in to the PPM Center database as a DBA and use the following command to flush the shared pool:

```
alter system flush shared_pool;
```

2. Download the Content Pack 2.0 upgrade bundle as follows:
 - a. Go to [Operational Reports Content for Project and Portfolio Management - Downloads](#) page on HPLN Web site (hp1n.hp.com/node/81/contentfiles).

Note: To access this Web site, you must provide your SAID for PPM Center.

- b. Select **PPM Operational Reporting CP2.0**.
 - c. Download the **CP2.0_Upgrade_Bundle.zip** file.
3. Extract the contents of the Operational Reporting for PPM Center Content Pack 2.0 upgrade bundle to its own directory (hereinafter referred to as the *<Op_Reports_Home>* directory).
 4. Make sure that an additional 2 GB is available on your C:\ drive for Windows installer. (Windows installer creates install patches under the C:\Windows\Installer folder.)

5. Install PPM Center version 9.12 or the later versions.

Note: If you have the required service agreement ID (SAID), you can get PPM Center software updates through the Software Update Manager (SUM) site (www1.itrc.hp.com/service/sum/home.do).

For information about how to download and install PPM Center 9.10 or the later versions, see the *Installation and Administration Guide* or the *Release Notes* for the later versions. You can obtain the *Release Notes* from the Software Product Manuals Web site (support.openview.hp.com/selfsolve/manuals).

6. Back up your Operational Reporting database.

Caution: Any customizations that you have made to HP-supplied reporting universes or preconfigured reports (for example, the Demand Versus Capacity report) on your existing Operational Reporting instance are lost during an upgrade.

Operational Reporting Upgrade for Windows Systems

This section includes instructions for upgrading an existing Operational Reporting deployment on a Windows system. Make sure that you are on Operational Reporting Content Pack 1.3 before the upgrade process.

Note: If, for some reason, you must stop the upgrade process, the upgrade will resume where it left off when you next start the upgrade. You can perform the upgrade as many times as necessary.

To upgrade Operational Reporting on a Windows system:

1. (Optional) Navigate to the `<Op_Report_Home>\DB\install\sample` directory and run the `sample_onetime_batch.bat` file to synchronize PPM database to Operational Reporting database.

Caution: You need to wait for the ETL to complete before proceeding with Step 2.

For details, see the Running Incremental ETL Jobs Manually section of the Operational Reporting Administrator's Guide.

2. Navigate to the `<Op_Reports_Home>\Sample` directory, run the `sample_preupgrade_rpt.bat` file.
3. The script performs a sanity check on PPM database. Do one of the following:

- If the sanity check fails, HP strongly recommends you to recreate the Operational Reporting database schema and reload PPM Center Data into the Operational Reporting database. For details, see ["Creating the Operational Reporting Database Schema" on page 15](#) and ["Loading PPM Center Data into the Operational Reporting Database" on page 23](#).
 - If the sanity check passes, continue with [Step 4](#).
4. Stop your PPM Servers.
 5. Grant necessary privileges to PPM Schema.

Connect to PPM database as SYSDBA and run the following SQL commands:

```
grant dba to <PPM_SCHEMA>;  
grant become user to <PPM_SCHEMA>;  
grant create view to <PPM_SCHEMA>;  
grant create sequence to <PPM_SCHEMA>;  
  
BEGIN  
    DBMS_STREAMS_AUTH.GRANT_ADMIN_PRIVILEGE(grantee => '<PPM_SCHEMA>',  
grant_privileges => true);  
END;
```

6. Navigate to the `<Op_Reports_Home>\Sample` directory, and open the `sample_upgrade_rpt.bat` file in a text editor.
7. In the PARAMETERS section, uncomment the parameter placeholders listed in the following table, and replace them with valid values.

| Parameter | Description |
|--------------------------|---|
| Reporting DB Schema Name | Operational Reporting database schema name Example value: RPT_SCHEMA Important: The Operational Reporting database schema name must be in all capital letters. |
| Reporting DB TNS Name | Identifies the Oracle instance that runs the Operational Reporting database schema. TNS name is configured in the <code>tnsnames.ora</code> file. Example value: RPT |

| Parameter | Description |
|--|--|
| PPM DB Schema Name | <p>PPM Center database schema name.</p> <p>This value should exist in the Oracle <code>tnsnames.ora</code> entry.</p> <p>Example value: PPM_SCHEMA</p> <p>Important: The PPM Center database schema name must be in all capital letters. If the name contains any lowercase characters, an error occurs.</p> |
| PPM DB TNS Name | <p>Oracle instance that runs the PPM Center database schema. TNS name is configured in the <code>tnsnames.ora</code> file.</p> |
| LOG mode | <p>Determines where log output goes.</p> <p>Valid values are FILE, DB, and BOTH.</p> <p>If set to FILE, the output goes into the <code>upgrade_rpt_<Date_Time>.log</code> file.</p> <p>If set to DB, the output goes into the database event log tables.</p> <p>If set to BOTH, the output goes into both the <code>upgrade_rpt_<Date_Time>.log</code> file and the database event log tables.</p> |
| Reporting DB data_ tablespace_name | <p>Name of the data tablespace for the Operational Reporting database</p> <p>Example value: RPT_DATA_TS</p> <p>Important: The Operational Reporting database data tablespace name must be in all capital letters.</p> |
| Reporting DB index_ tablespace_name | <p>Name of the index tablespace for the Operational Reporting database</p> <p>Example value: RPT_INDEX_TS</p> <p>Important: The Operational Reporting database index tablespace name must be in all capital letters.</p> |
| Reporting DB DATA_ NOLOGGING_ TABLESPACE | <p>Separate tablespace that requires no redo log for the Operational Reporting database to store data.</p> <p>Example value: PPM_DATA_TS_NL</p> |

| Parameter | Description |
|--|--|
| Reporting DB INDEX_NOLOGGING_TABLESPACE | Separate tablespace that requires no redo log for the Operational Reporting database to store indexes. Example value: RPT_INDEX_TS_NL |
| PPM DB data_tablespace_name | PPM Center database data tablespace name Note: This refers to the existing data tablespace in the <i>PPM Center database schema</i> . The PPM Center schema stores this in the KINS_TABLESPACES table. Example value: PPM_DATA_TS Important: The PPM Center database data tablespace name must be in all capital letters. |
| PPM DB index_tablespace_name | PPM Center database index tablespace name Note: This refers to the existing index tablespace in the <i>PPM Center database schema</i> . The PPM Center schema stores this in the KINS_TABLESPACES table. Example value: PPM_INDEX_TS Important: The PPM Center database index tablespace name must be in all capital letters. |
| PPM Server Status, PPM_DOWN_NO, PPM_DOWN_YES | If set to PPM_DOWN_NO in the <code>sample_resync_ppm.bat</code> or <code>sample_upgrade_rpt.bat</code> file, then when the script is run, performs a check to determine whether any PPM Servers are running. If any node is running, the upgrade stops so that you can shut down all running nodes. If set to PPM_DOWN_YES, the PPM Server check is not performed. |
| DB_LINK_NAME to PPM | Name of the link to the PPM Center database This value is generated in the Operational Reporting database schema. Example value: PPM_DB_LINK Important: The name of the link to the PPM Center database must be in all capital letters. |

8. Run `sample_upgrade_rpt.bat`.
9. Review the generated `upgrade_rpt_<Date_Time>.log` file (located in the `<Op_Reports_`

Home>\log directory). Check whether there is an error Error:ORA- in the log file or any error messages at the end of log file. If no such errors occur, upgrade is successful.

10. Revoke system privileges from PPM Schema.

Connect to PPM database as SYSDBA and run the following commands:

```
revoke dba from <PPM_SCHEMA>  
revoke become user from <PPM_SCHEMA>;
```

11. Restart the PPM Servers, one at a time.
12. To verify a successful upgrade of Operational Reporting, run the query for an HP-supplied report. For information about how to run HP-supplied operational reports, see the *Operational Reporting User's Guide*.

Recovering from an Upgrade Failure

An upgrade to PPM Center Content Pack 2.0 can potentially fail for several reasons. The process stops if, for example, the connection to a remote database is lost, the client machine running an upgrade script goes down, or if PPM Center data are missing. This section provides information about what to do if your upgrade fails during the different stages of the upgrade process.

Upgrade Failure Resulting from Active PPM Servers

The `sample_resync_ppm.bat` and `sample_upgrade_rpt.bat` files both include the PPM Server Status parameter, which can be set to either `PPM_DOWN_NO` or `PPM_DOWN_YES`. If you set the PPM Server Status parameter to `PPM_DOWN_NO` in either of these files and then execute the file, the upgrade checks to determine whether any PPM Servers are active. If an active node is detected during the run, the upgrade stops and the following message is displayed:

```
Failed with this error => PPM DOWN is required. One or more PPM Servers is active.  
If all nodes are down, pass PPM_DOWN_YES, *** aborting upgrade...
```

If this error occurs, do the following:

1. Shut down every active node in the server cluster.
2. Open the sample batch or shell script (either `sample_upgrade_rpt.bat` or `sample_resync_ppm.bat`) and change the PPM Server Status parameter value from `PPM_DOWN_NO` to `PPM_DOWN_YES`.
3. Run the script again.

The upgrade process skips the PPM Server check after you set PPM Server Status parameter value to `PPM_DOWN_YES`.

Failure During the Sample_preupgrade_rpt.bat Run

If the upgrade fails while the Sample_preupgrade_rpt.bat script is running, do the following:

1. Review the generated preupgrade_rpt_<Date_Time>.log file (located in the <Op_Reports_Home>\log folder).
2. Correct any reported errors, and then run the Sample_preupgrade_rpt.bat script again.

Failure During the Sample_upgrade_rpt.bat Run

If the upgrade fails while the Sample_upgrade_rpt.bat script is running, do the following:

1. Review the generated upgrade_rpt_<Date_Time>.log file (located in the <Op_Reports_Home>\log folder).
2. Correct any reported errors, and then run the Sample_upgrade_rpt.bat script again.

Chapter 5: Upgrading Operational Reporting on a UNIX System

Preparing to Upgrade

This section addresses the tasks to complete before you begin to upgrade to PPM Center version Content Pack 2.0.

As an existing user of the Operational Reporting solution, you can upgrade the Operational Reporting only from Content Pack 1.3 to Content Pack 2.0. For instructions about upgrading Operational Reporting solution to Content Pack 1.3, see *Operational Reporting Administrator's Guide* for Content Pack 1.3.

Note: If you are deploying the Operational Reporting solution for the first time, follow the instructions for deployment provided in "[Deploying Operational Reporting on UNIX Systems](#)" on [page 27](#).

To prepare to upgrade Operational Reporting:

1. Log in to the PPM Center database as a DBA and use the following command to flush the shared pool:

```
alter system flush shared_pool;
```

2. Download the Content Pack 2.0 upgrade bundle as follows:
 - a. Go to [Operational Reports Content for Project and Portfolio Management - Downloads](#) page on HPLN Web site (hp1n.hp.com/node/81/contentfiles).

Note: To access this Web site, you must provide your SAID for PPM Center.

- b. Select **PPM Operational Reporting CP2.0**.
 - c. Download the **CP2.0_Upgrade_Bundle.tar.gz** file.
3. Extract the contents of the Operational Reporting for PPM Center Content Pack 2.0 upgrade bundle to its own directory (hereinafter referred to as the *<Op_Reports_Home>* directory).
 4. Install PPM Center version 9.12 or the later versions.

Note: If you have the required service agreement ID (SAID), you can get PPM Center software updates through the Software Update Manager (SUM) site (www1.itrc.hp.com/service/sum/home.do).

For information about how to download and install PPM Center and the later versions, see the *Installation and Administration Guide* or the *Release Notes* for the later versions. You can obtain the *Release Notes* from the Software Product Manuals Web site (support.openview.hp.com/selfsolve/manuals).

5. Back up your Operational Reporting database.

Caution: Any customizations that you have made to HP-supplied reporting universes or preconfigured reports (for example, the Demand Versus Capacity report) on your existing Operational Reporting instance are lost during an upgrade.

Operational Reporting Upgrade for UNIX Systems

This section includes instructions for upgrading an existing Operational Reporting deployment on a UNIX system. Make sure that you are on Operational Reporting Content Pack 1.3 before the upgrade process.

Note: If, for some reason, you must stop the upgrade process, the upgrade process will resume where it left off when you next start the upgrade. You can perform the upgrade as many times as necessary.

To upgrade Operational Reporting on a Unix system:

1. *Optional:* Navigate to the `<Op_Report_Home>/DB/install/sample` directory and run the `sample_onetime_batch.sh` file to synchronize PPM database to Operational Reporting database.

Caution: You need to wait for the ETL to complete before proceeding with Step 2.

For details, see the Running Incremental ETL Jobs Manually section of the Operational Reporting Administrator's Guide.

2. Navigate to the `<Op_Reports_Home>/Sample` directory, run the `sample_preupgrade_rpt.sh` file.
3. The script performs a sanity check on PPM database. Do one of the following:

- If the sanity check fails, HP strongly recommends you to recreate the Operational Reporting database schema and reload PPM Center Data into the Operational Reporting database. For details, see ["Creating the Operational Reporting Database Schema" on page 27](#) and ["Loading PPM Center Data into the Operational Reporting Database" on page 36](#).
 - If the sanity check passes, continue with [Step 4](#).
4. Stop your PPM Servers.
 5. Grant necessary privileges to PPM Schema.

Connect to PPM database as SYSDBA and run the following SQL commands:

```
grant dba to <PPM_SCHEMA>;
grant become user to <PPM_SCHEMA>;
grant create view to <PPM_SCHEMA>;
grant create sequence to <PPM_SCHEMA>;

BEGIN
    DBMS_STREAMS_AUTH.GRANT_ADMIN_PRIVILEGE(grantee => '<PPM_SCHEMA>',
    grant_privileges => true);
END;
```

6. Navigate to the `<Op_Reports_Home>/Sample` directory, and open the `sample_upgrade_rpt.sh` file in a text editor.
7. In the PARAMETERS section, uncomment the parameter placeholders listed in the following table, and replace them with valid values.

| Parameter | Description |
|--------------------------|---|
| Reporting DB Schema Name | Operational Reporting database schema name Example value: RPT_SCHEMA Important: The Operational Reporting database schema name must be in all capital letters. |
| Reporting DB TNS Name | Identifies the Oracle instance that runs the Operational Reporting database schema. TNS name is configured in the <code>tnsnames.ora</code> file. Example value: RPT |

| Parameter | Description |
|--|--|
| PPM DB Schema Name | <p>PPM Center database schema name.</p> <p>This value should exist in the Oracle <code>tnsnames.ora</code> entry.</p> <p>Example value: PPM_SCHEMA</p> <p>Important: The PPM Center database schema name must be in all capital letters. If the name contains any lowercase characters, an error occurs.</p> |
| PPM DB TNS Name | <p>Oracle instance that runs the PPM Center database schema. TNS name is configured in the <code>tnsnames.ora</code> file.</p> |
| LOG mode | <p>Determines where log output goes.</p> <p>Valid values are FILE, DB, and BOTH.</p> <p>If set to FILE, the output goes into the <code>upgrade_rpt_<Date_Time>.log</code> file.</p> <p>If set to DB, the output goes into the database event log tables.</p> <p>If set to BOTH, the output goes into both the <code>upgrade_rpt_<Date_Time>.log</code> file and the database event log tables.</p> |
| Reporting DB data_ tablespace_name | <p>Name of the data tablespace for the Operational Reporting database</p> <p>Example value: RPT_DATA_TS</p> <p>Important: The Operational Reporting database data tablespace name must be in all capital letters.</p> |
| Reporting DB index_ tablespace_name | <p>Name of the index tablespace for the Operational Reporting database</p> <p>Example value: RPT_INDEX_TS</p> <p>Important: The Operational Reporting database index tablespace name must be in all capital letters.</p> |
| Reporting DB DATA_ NOLOGGING_ TABLESPACE | <p>Separate tablespace that requires no redo log for the Operational Reporting database to store data.</p> <p>Example value: PPM_DATA_TS_NL</p> |

| Parameter | Description |
|--|---|
| Reporting DB INDEX_ NOLOGGING_ TABLESPACE | <p>Separate tablespace that requires no redo log for the Operational Reporting database to store indexes.</p> <p>Example value: RPT_INDEX_TS_NL</p> |
| PPM DB data_ tablespace_name | <p>PPM Center database data tablespace name</p> <p>Note: This refers to the existing data tablespace in the <i>PPM Center database schema</i>. The PPM Center schema stores this in the KINS_ TABLESPACES table.</p> <p>Example value: PPM_DATA_TS</p> <p>Important: The PPM Center database data tablespace name must be in all capital letters.</p> |
| PPM DB index_ tablespace_name | <p>PPM Center database index tablespace name</p> <p>Note: This refers to the existing index tablespace in the <i>PPM Center database schema</i>. The PPM Center schema stores this in the KINS_ TABLESPACES table.</p> <p>Example value: PPM_INDEX_TS</p> <p>Important: The PPM Center database index tablespace name must be in all capital letters.</p> |
| PPM Server Flag, PPM_DOWN_NO, PPM_DOWN_YES | <p>If set to PPM_DOWN_NO in the sample_resync_ppm.sh or sample_upgrade_rpt.sh file, then when the script is run, performs a check to determine whether any PPM Servers are running. If any node is running, the upgrade stops so that you can shut down all running nodes. If set to PPM_DOWN_YES, the PPM Server check is not performed.</p> |
| DB_LINK_NAME to PPM | <p>Name of the link to the PPM Center database</p> <p>This value is generated in the Operational Reporting database schema.</p> <p>Example value: PPM_DB_LINK</p> <p>Important: The name of the link to the PPM Center database must be in all capital letters.</p> |

8. Run `sample_upgrade_rpt.sh`.
9. Review the generated `upgrade_rpt_<Date_Time>.log` file (located in the `<Op_Reports_`

Home>/log directory). Check whether there is an error Error:ORA- in the log file or any error messages at the end of log file. If no such errors occur, upgrade is successful.

10. Revoke system privileges from PPM Schema.

Connect to PPM database as SYSDBA and run the following commands:

```
revoke dba from <PPM_SCHEMA>  
revoke become user from <PPM_SCHEMA>;
```

11. Restart the PPM Servers, one at a time.
12. To verify a successful upgrade of Operational Reporting, run the query for an HP-supplied report. For information about how to run HP-supplied operational reports, see the *Operational Reporting User's Guide*.

Recovering from an Upgrade Failure

An upgrade to PPM Center Content Pack 2.0 can potentially fail for several reasons. The process will stop if, for example, the connection to a remote database is lost, the client machine running an upgrade script goes down, or if PPM Center data are missing. This section provides information about what to do if your upgrade fails during the different stages of the upgrade process.

Upgrade Failure Resulting from Active PPM Servers

The `sample_resync_ppm.sh` and `sample_upgrade_rpt.sh` files both include the PPM Server Status parameter, which can be set to either `PPM_DOWN_NO` or `PPM_DOWN_YES`. If you set the PPM Server Status parameter to `PPM_DOWN_NO` in either of these files and then execute the file, the upgrade checks to determine whether any PPM Servers are active. If an active node is detected during the run, the upgrade stops and the following message is displayed:

Failed with this error => PPM DOWN is required. One or more PPM Servers is active. If all nodes are down, pass PPM_DOWN_YES, *** aborting upgrade...

If this error occurs, do the following:

1. Shut down every active node in the server cluster.
2. Open the sample batch or shell script (either `sample_upgrade_rpt.sh` or `sample_resync_ppm.sh`) and change the PPM Server Status parameter value from `PPM_DOWN_NO` to `PPM_DOWN_YES`.
3. Run the script again.

The upgrade process skips the PPM Server check after you set the PPM Server Status parameter value to `PPM_DOWN_YES`.

Failure During the `sample_preupgrade_rpt.sh` Run

If the upgrade fails while the `sample_preupgrade_rpt.sh` script is running, do the following:

1. Navigate to the `<Op_Reports_Home>/log` directory and review the generated `preupgrade_rpt.log` file.
2. Correct any reported errors, and then run the `Sample_preupgrade_rpt.sh` script again.

Failure During the `sample_upgrade_rpt.sh` Run

If the upgrade fails while the `sample_upgrade_rpt.sh` script is running, just run the script again when appropriate.

Chapter 6: Creating Views for Request Types and Table Components

Operational Reporting for PPM Center creates views for request types and table components. A request type or a table component is mapped to a view in Operational Reporting. The view contains all the data of the request type or the table component, including custom data.

The name of the view created for a request type is in the format: RTP_REQ_<request_type>, while the name of the view created for a table component is in the format: RTP_TC_<table_component>.

Example: View created for request type

The request type Bug have the following custom fields:

| Prompt | Token |
|----------------------------|------------|
| Module | MODULE |
| Difficulty | DIFFICULT |
| Platform | PLATFORM |
| Estimated Time to Complete | COMP_TYPE |
| Impact | IMPACT |
| Error Log | ERROR_LOG |
| Reproducible | REPRO |
| Resolution | RESOLUTION |
| Steps To Replicate | REPLICATE |
| Duplicate ID | DUP_ID |
| Resolution Summary | RESOL_SUM |

Operational Reporting creates the view RPT_REQ_BUG for this request type as follows:

```
RPT_REQ_BUG  
(  
  REQUEST_ID,  
  REQUEST_TYPE_ID,  
  MODULE,  
  PLATFORM,
```

```
IMPACT,  
ERROR_LOG,  
REPRO,  
REPLICATE,  
DIFFICULT,  
COMP_TYPE,  
RESOLUTION,  
DUP_ID,  
RESOL_SUM  
)
```

Example: View created for table component

The table component FINANCIAL_SCOPE has the following custom fields:

| Prompt | Token |
|--------------------|--------------------|
| Raised by | RAISED_BY |
| Status | STATUS |
| Change Category | CHANGE_CATEGORY |
| Change Description | CHANGE_DESCRIPTION |
| Cost of Impact | COST_IMPACT |
| Estimated Effort | EFFORT_DAYS |
| Create Date | CREATE_DATE |

Operational Reporting creates the view RPT_TC_FINANCIAL_SCOPE for this table component as follows:

```
RPT_TC_FINALCIAL_SCOPE  
(  
  REQUEST_ID,  
  PARAMETER_TOKEN,  
  ROW_SEQUENCE_NUMBER,  
  RAISED_BY,  
  STATUS,  
  CHANGE_CATEGORY,  
  CHANGE_DESCRIPTION,  
  COST_IMPACT,  
  EFFORT_DAYS,  
  CREATE_DATE  
)
```

Chapter 7: Refreshing Operational Reporting Data

Synchronizing Data in the Operational Reporting and PPM Center Database Schema

This chapter provides information about how data in the PPM Center database schema and the Operational Reporting database schema are synchronized.

Running Incremental ETL Jobs

The load script that you run during Operational Reporting deployment performs a full ETL to load all PPM Center data into the Operational Reporting database schema. Incremental ETL jobs are scheduled to run automatically every 24 hours thereafter. These incremental ETL jobs cover the day-to-day updates for the PPM Center data tables.

The incremental ETL job that runs automatically every 24 hours is named PPM_ETL_BATCH_JOB. You can use an Oracle command to reschedule or change the frequency of the PPM_ETL_BATCH_JOB run. (For information on how to reschedule the PPM_ETL_BATCH_JOB, see the Oracle Database Online Documentation.) This section contains instructions on how to run incremental ETL jobs manually.

Checking ETL Job Progress

To check the job progress, you can query the RPT_EVENT_LOG_DETAIL table, as follows:

```
select event_time, lead(event_time,1) over
(order by event_log_id desc), round((event_time - lead(event_
time,1) over (order by event_log_id desc))*24*60 ,2) durations,
t1.*
from rpt_event_log_detail t1
order by event_log_id desc;
```

where the value of event_time must contain both date value and time value.

If the value of event_time does not contain date value or time value, configure the settings in Oracle client tool as follows:

1. Open Oracle SQL Developer.
2. Click **Tools > Preferences**.
3. In the left pane, expand **Database** (if it is not already expanded).
4. Under **Database**, click **NLS**.
5. Set the date format to: YYYY-MM-DD HH24:MI:SS
6. Click **OK**.

To view the status of an incremental ETL job, you can query the job control tables (RPT_ETL_JOB and RPT_EVENT_LOG_DETAIL tables).

Verifying Successful Incremental ETL Jobs

To determine whether the last incremental ETL job run completed successfully, run the following:

```
select event_time, lead(event_time,1) over
(order by event_log_id desc), round((event_time - lead(event_
time,1) over (order by event_log_id desc))*24*60 ,2) durations,
t1.*
from rpt_event_log_detail t1
order by event_log_id desc;

SELECT * FROM rpt_etl_job ORDER BY etl_job_id desc;
```

where the value of event_time must contain both date value and time value.

If the value of event_time does not contain date value or time value, configure the settings in Oracle client tool. For detailed steps, see ["If the value of event_time does not contain date value or time value, configure the settings in Oracle client tool as follows: "](#) on the previous page.

Note: HP recommends that you delete the contents of the rpt_event_log_detail and rpt_etl_job order tables at least once a month to prevent them from becoming too large. You must delete the contents manually.

Running Incremental ETL Jobs Manually

To manually run an incremental ETL batch job immediately:

1. Navigate to the <Op_Report_Home>/DB/install/sample directory.
2. Open the sample_onetime_batch.bat file (or sample_onetime_batch.sh) file in a text editor, and then replace the parameter placeholders with valid values, as shown in the following table.

| Parameter | Value |
|--------------------------|--|
| Reporting DB Schema Name | Operational Reporting database schema name |
| Reporting DB TNS Name | Operational Reporting database TNS name |
| ETL BATCH JOB NAME | Any job name Example: TM_ETL_DAILY |

3. Run the sample_onetime_batch.bat (or sample_onetime_batch.sh) file.

Example:

```
call sample_onetime_batch.bat REPORTING SCHEMA ORASID TEST_ETL_JOB
```

4. When you are prompted, type the password for the Operational Reporting database schema.
5. To check the job progress, you can query the RPT_EVENT_LOG_DETAIL table, as follows:

```
select event_time, lead(event_time,1) over  
(order by event_log_id desc), round((event_time -  
lead(event_time,1) over (order by event_log_id desc))*24*60  
,2) durations, t1.*  
from rpt_event_log_detail t1  
order by event_log_id desc;
```

where the value of event_time must contain both date value and time value.

If the value of event_time does not contain date value or time value, configure the settings in Oracle client tool. For detailed steps, see ["If the value of event_time does not contain date value or time value, configure the settings in Oracle client tool as follows: " on page 55.](#)

To view the status of an incremental ETL job, you can query the job control tables (RPT_ETL_JOB and RPT_EVENT_LOG_DETAIL tables).

Note: If an incremental ETL job fails, it is rerun automatically when the Oracle scheduler starts the job for the next scheduled run, or when you run the job manually, whichever occurs first.

Purging Data

In PPM schema, we capture and store change data in `<ppm_table>_cdc` tables. We use the `DBMS_SCHEDULER` package to create a purge job named `cdc$_default_purge_job`. This purge job calls the `PPM_CDC.PURGE_CDC` procedure to remove data that has been processed by incremental ETL. By default, `cdc$_default_purge_job` runs every 24 hours. You can reschedule the purge job using `DBMS_SCHEDULER.SET_ATTRIBUTE` and setting the `repeat_interval` attribute.

Running the `cdc$_default_purge_job` regularly ensures that the tables do not grow without limit. If you have a large volume of data and need to schedule frequent incremental ETL jobs, you can schedule the `cdc$_default_purge_job` to run more frequently than the default of every 24 hours.

For information about the `DBMS_SCHEDULER` package, see the [Oracle Database PL/SQL Packages and Types Reference](#).

Note: Do not manually delete data in `<ppm_table>_cdc` tables. Always call the `PURGE_CDC` procedure to remove them, otherwise, the data in PPM schema and reporting schema are not synchronized.

PPM Center Data Transfer During ETL

The following sections describe how PPM Center data are transferred during the ETL process.

Common Dimension Data Transfer

The COMMON ETL job transfers all PPM Center data (that fall within the interval defined by the specified start and end dates) and that are shared by HP Time Management, HP Resource Management, and HP Financial Management into the Operational Reporting database schema.

HP Demand Management Data Transfer

A full ETL for HP Demand Management (DM) transfers all project data (that fall within the interval defined by the specified start and end dates) from the HP Demand Management tables into the Operational Reporting database schema. A subsequent incremental DM ETL job loads HP Demand Management data that have changed since the last DM ETL job run.

HP Portfolio Management Data Transfer

A full ETL for HP Portfolio Management (PfM) transfers all project data (that fall within the interval defined by the specified start and end dates) from the HP Portfolio Management tables into the

Operational Reporting database schema. A subsequent incremental PfM ETL job loads HP Portfolio Management data that have changed since the last PfM ETL job run.

HP Project Management Data Transfer

A full ETL for HP Project Management (PM) transfers all project data (that fall within the interval defined by the specified start and end dates) from the HP Project Management tables into the Operational Reporting database schema. A subsequent incremental PM ETL job loads HP Project Management data that have changed since the last PM ETL job run.

HP Time Management Data Transfer

A full ETL for HP Time Management (TM) transfers all time-sheet data (that fall within the interval defined by the specified start and end dates) from the HP Time Management tables into the Operational Reporting database schema. A subsequent incremental TM ETL job loads HP Time Management data that have changed since the last TM ETL job run.

HP Resource Management Data Transfer

A full ETL for HP Resource Management (RM) transfers all resource capacity, demand, and actual effort data (that fall within the interval defined by the specified start and end dates) from the HP Resource Management tables into the Operational Reporting database schema. A subsequent incremental RM ETL job loads HP Resource Management data that have changed since the last RM ETL job run.

HP Financial Management Data Transfer

The FM incremental ETL job transfers all HP Financial Management data (that fall within the interval defined by the specified start and end dates) from the HP Financial Management tables into the Operational Reporting database schema. A subsequent incremental FM ETL job loads HP Financial Management data changed since the last FM ETL job run.

Date Range for Transferred Data

The date range for the data moved to the Operational Reporting database during the ETL process is determined by the values you specify for the ETL_START_DATE and ETL_END_DATE parameters. The start date of the fiscal year is determined the year you specify for the ETL_START_DATE value. ["Table 7-1. Effect of PPM Center fiscal year on the calculated ETL start date" on the next page](#) shows how this influences the actual start date for the ETL.

Table 7-1. Effect of PPM Center fiscal year on the calculated ETL start date

| Specified ETL_START_DATE | Start Date for the PPM Center Fiscal Year | Calculated (Actual) ETL Start Date |
|--------------------------|---|------------------------------------|
| 01/15/2008 | January 1 | 01/01/2008 |
| | November 1 | 11/01/2007 |

The end date of the fiscal year is determined the year you specify for the ETL_END_DATE value. ["Table 7-2. Effect of PPM Center fiscal year on the calculated ETL end date"](#) below shows how this influences the actual end date for the ETL.

Table 7-2. Effect of PPM Center fiscal year on the calculated ETL end date

| Specified ETL_END_DATE | End Date for the PPM Center Fiscal Year | Calculated (Actual) ETL End Date |
|------------------------|---|----------------------------------|
| 11/30/2012 | December 31 | 12/31/2012 |
| | October 31 | 10/31/2013 |

Date Range for Transferred HP Time Management Data

For HP Time Management data, the value set for the ETL_START_DATE parameter determines which time sheets' data are brought into the Operational Reporting database. If a time sheet has an end date that is on or later than the ETL_START_DATE, then that time sheet is used to generate data in the Operational Reporting schema.

The ETL_END_DATE parameter value is not used. Except for cancelled time sheets, all time sheets with end dates that fall after the ETL start date are brought over.

Date Range for Transferred HP Resource Management Data

Calculated ETL start and end dates affect HP Resource Management data transfer in the following ways:

- Fiscal period definitions are brought over for fiscal periods whose start dates and end dates fall within the time period specified by the calculated start and end dates.
- Resource demand data are brought over for all staffing profiles whose demand falls within the time period defined by the calculated start and end dates.
- Resource capacity data are brought over for all of the resources for the time period between the calculated start and end dates, provided that the resource's end date is later than the calculated end date, and the resource's start date falls within the time period specified by the calculated start and end dates.

- Resource actual effort data are brought over for all the time sheets (excluding cancelled time sheets) with ending dates later than the calculated start date.

Caution: If you have long-running projects, keep in mind that requests created before the ETL start date you specify are not brought over, and so the actual effort data for these requests are not available for reporting.

Extending the Time Range of Resource Capacity Data

The Resource Capacity data for resources that do not have an end date are generated based on the ETL start and end dates during the initial load. You can use the extend data script (`sample_extend_data.bat` or `sample_extend_data.sh`) to extend this time interval so that you can compare resource capacity and demand over time. Suppose, for example, that the last full ETL populated the Operational Reporting database with data through 2011. You can use the extend data script to include data for additional years, for example, through the calendar year two years in the future.

The start date for the data loaded using the extend data script is the day after the end year boundary. The end year boundary is based on the end date that you specify and the fiscal calendar's year end. (See "[Table 7-1. Effect of PPM Center fiscal year on the calculated ETL start date](#)" on the previous page and "[Table 7-2. Effect of PPM Center fiscal year on the calculated ETL end date](#)" on the previous page.)

The extend data script runs the full ETL for capacity and demand for the extended time period and performs the incremental ETL for COMMON, RM, FM, and TM universes. If an incremental ETL job started by the extend script fails, you must run the incremental ETLs again. There is no need to run the extend data script again.

Recommendations for Running the Extend Data Script

To minimize the performance impact of running the extend data script, consider the following:

- Specify a data extension of just one year at a time instead of specifying multiple years.
- When you run `sample_extend_data.bat`, the script first drops all of the bitmap indexes in the HP Resource Management fact tables, and then recreates the indexes after loading the data. HP recommends that you *not* run reports during the extend data script run.

To run the extend data script, do the following:

1. Gather the information listed in the following table.

| Variable in the Extend Data Script | Description |
|------------------------------------|---|
| Reporting DB Schema Name | Operational Reporting database schema name |
| Reporting DB TNS Name | Operational Reporting database TNS name |
| Reporting DB index_tablespace_name | Name of the index tablespace for the Operational Reporting database |
| ETL end date (mm-dd-yyyy) | End date for the PPM Center data to extract, transform, and load into the Operational Reporting database schema. ^a |

a. HP recommends you not to set the value for the ETL_END_DATE parameter too far away from the current fiscal year, otherwise performance issues may occur in incremental ETL. You must also extend the data and date to that before the ETL_END_DATE value; otherwise the data in the RPT_FCT_RM_RESOURCE_EFFORT table might be lost.

2. Log on to the BusinessObjects server machine.
3. Do one of the following:
 - On a Windows system, navigate to the <Op_Reports_Home>\Sample directory and open the sample_extend_data.bat file in a text editor.
 - On a UNIX system, navigate to the <Op_Reports_Home>/Sample directory and open the sample_extend_data.sh file in a text editor.
4. Replace each of the variables in the extend data script with the values you prepared for [Step 1](#), and then save and close the file.
5. Depending on your operating system, do one of the following:
 - On a Windows system, run sample_extend_data.bat.
 - On a UNIX system, run sample_extend_data.sh.
6. During the extend data script run, provide the Operational Reporting database schema password when prompted.
7. The script creates the extend_data.log file in the <Op_Report_Home>/DB/install/log directory. Log data are also captured in the RPT_EVENT tables. Review the log files and data.

Figure: Forecast and actual

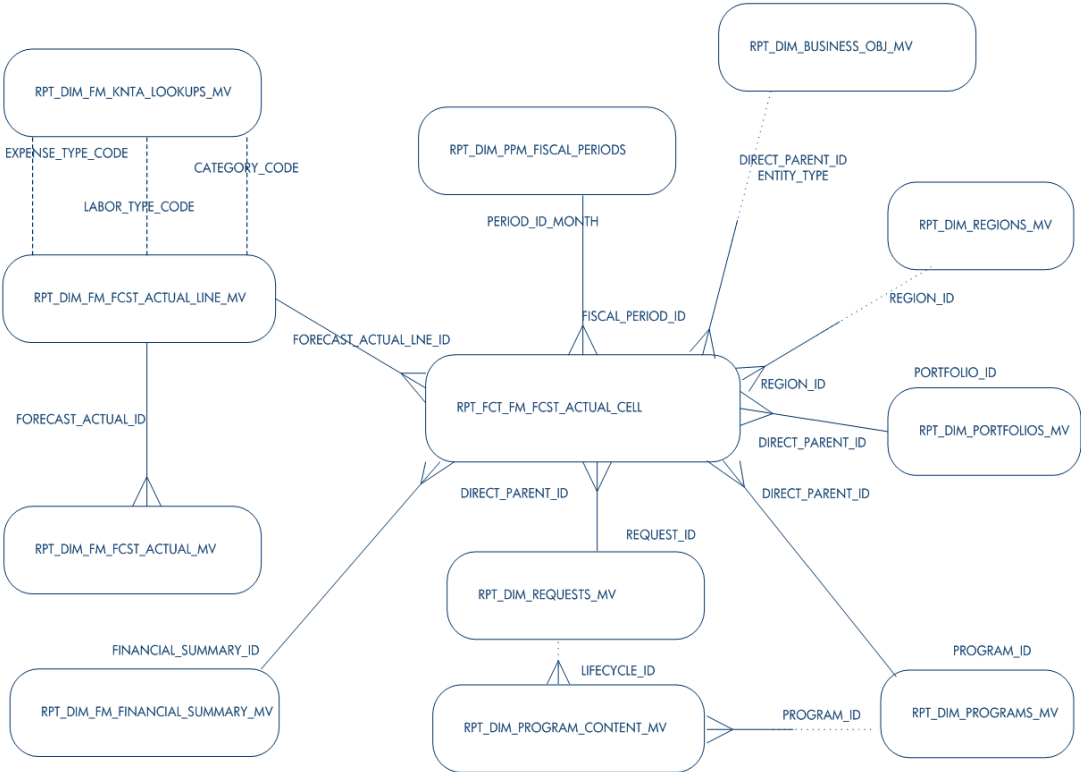


Figure: Forecast, actual, and approved budget

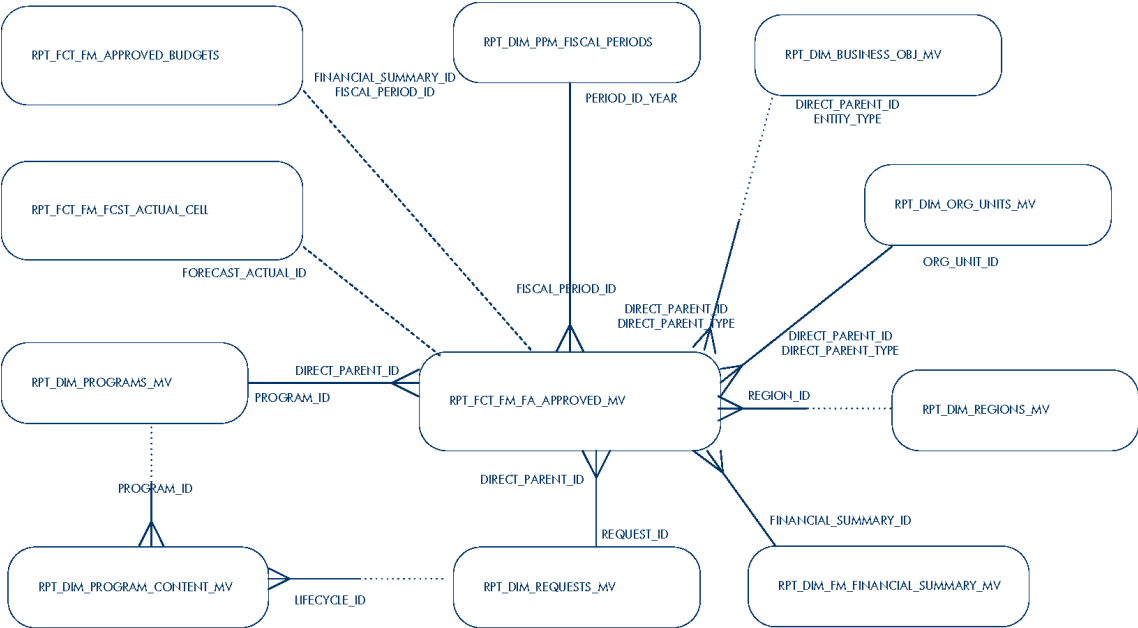


Figure: PM Universe



Figure: Resource capacity

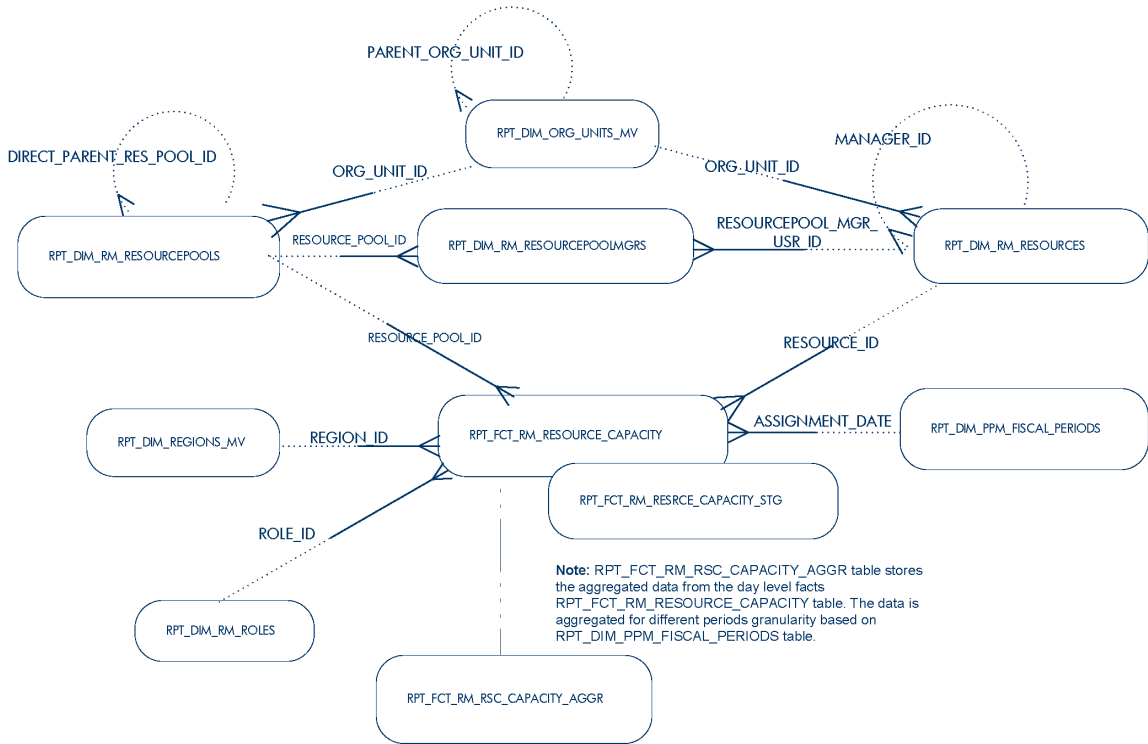


Figure: Resource demand

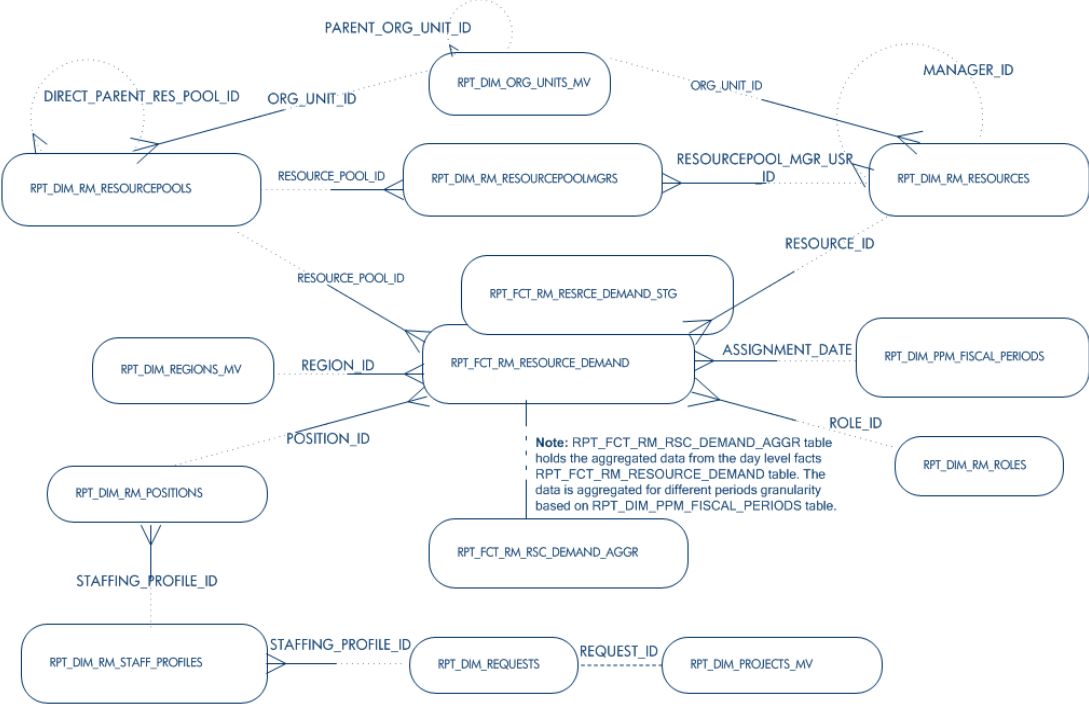


Figure: Resource effort

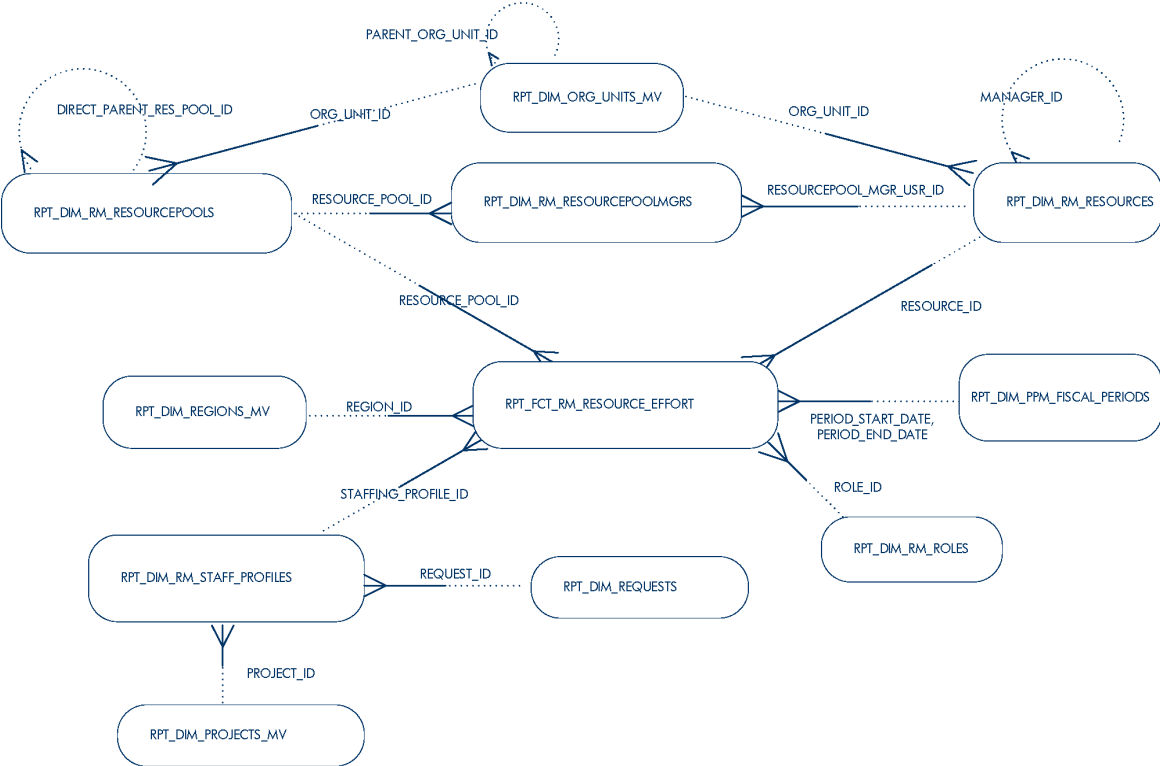


Figure: Time management

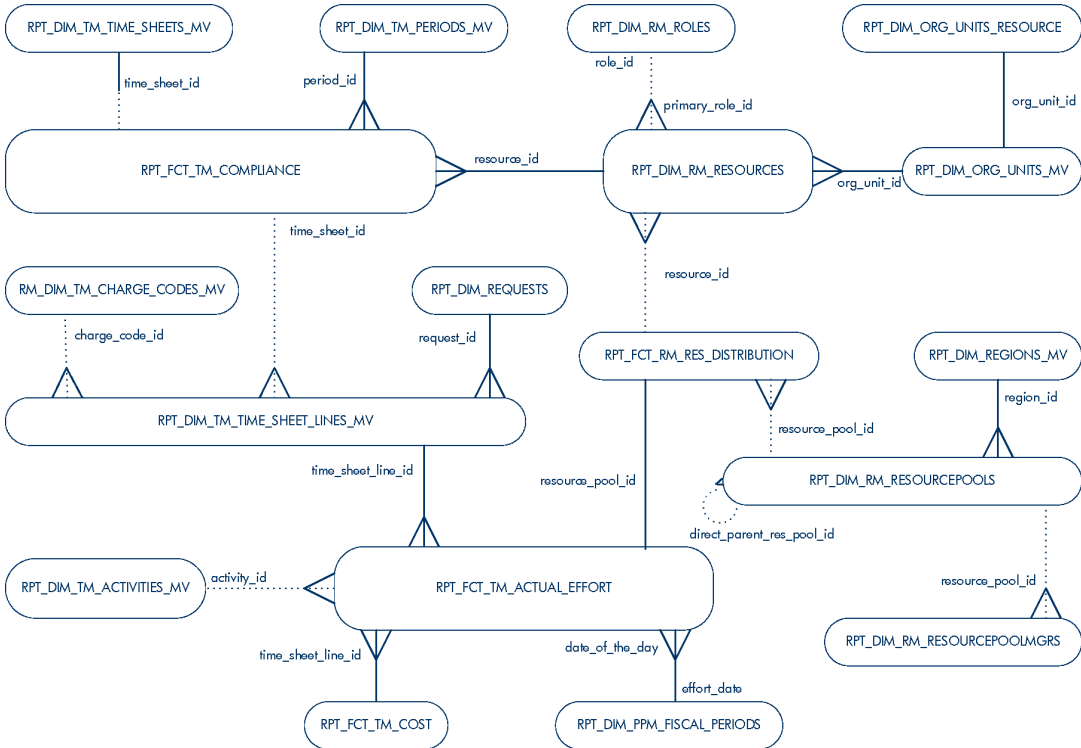
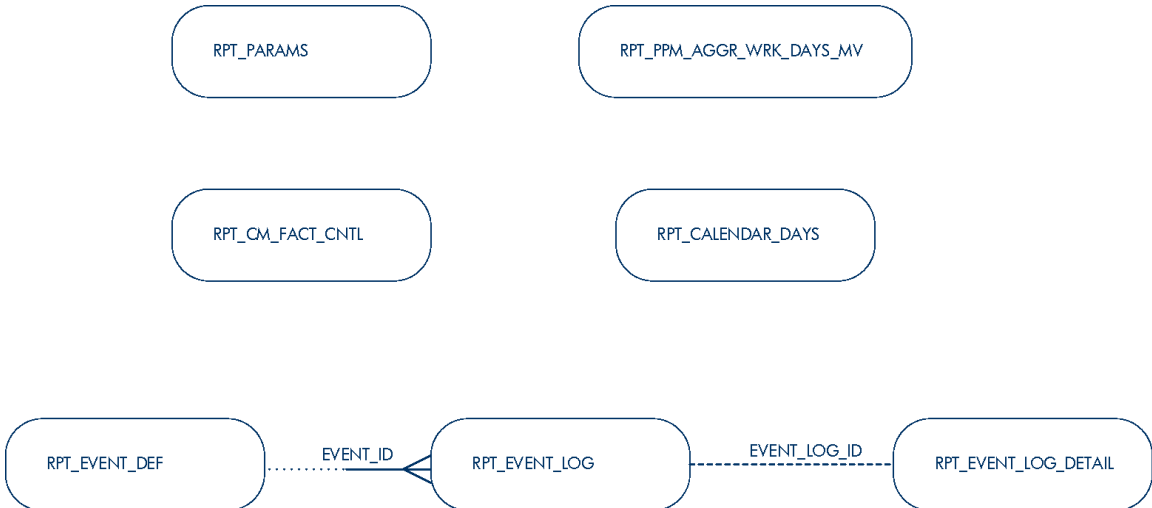


Figure: Other tables used during processing



Chapter 9: Definitions of Operational Reporting Tables

This chapter contains a detailed description of each table in the Operational Reporting schema. Every table definition provides a description of the table, a list of all foreign keys, indexes, and sequences for the table, as well as a description of each column in the table.

You can refer to this chapter to obtain a detailed understanding of the underlying structure of Operational Reporting.

RPT_CALENDAR_DAYS

This table is a single column table that is populated with dates during etl load_data. The dates are in the range of start boundary and end boundary for etl. This table is used during the full etl logic to improve the performance.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-----------------|----------|-----------|---|
| ASSIGNMENT_DATE | NULLABLE | DATE | Date field that falls between the start and the end boundaries of full etl. |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|----------------------|------------|----------|-----------------|
| RPT_CALENDAR_DAYS_UI | UNIQUE | 1 | ASSIGNMENT_DATE |

Sequences

This table uses no sequences.

RPT_CM_FACT_CNTL

This table is a single column table, that is used during the Common incremental etl which process the changes occurred in common dimensions. It stores the name of the table that currently is being processed for the changes occurred in PPM. Reading this table the switchable views builds the view to get data from the original PPM table or only the changes made on the PPM table via CDC table. When there is nothing to process, it is populated with a dummy number.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-------------|----------|-----------|--|
| CDC | NULLABLE | CHAR | Contains the table name that is currently being processed from the subscribed list of PPM tables for change capture. |

Indexes

This table uses no indexes.

Sequences

This table uses no sequences.

RPT_DEPLOYMENT

This table holds deployment history. Each time user successfully upgrade, it adds an entry in this table.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|---------------|----------|--------------|--|
| DEPLOYMENT_ID | NOT NULL | VARCHAR2(40) | Uniquely identify a deployment record |
| VERSION | NOT NULL | VARCHAR2(10) | Version of the product to be installed |
| DEPLOYED_DATE | NULLABLE | DATE | Date when upgrade is installed |
| DESCRIPTION | NULLABLE | VARCHAR2(80) | Brief description of the upgrade |

Indexes

This table uses no indexes.

Sequences

This table uses no sequences.

RPT_DIM_BUSINESS_OBJ_MV

This is a materialized view representing the Business Objective dimension in the Star schema.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-----------------------|----------|-----------|---|
| BUSINESS_OBJECTIVE_ID | NULLABLE | NUMBER | Represents the unique identifier of Business Objective and is derived from PPM KPMO_BUSINESS_OBJECTIVES_NLS table |

| Column Name | Null? | Data Type | Description |
|-------------------------|----------|---------------|--|
| BUSINESS_OBJECTIVE_NAME | NULLABLE | VARCHAR2 (80) | Name of the business objective |
| ENTITY_ID | NULLABLE | NUMBER | Represents the ID of the Program or Request that is associated with the business objective |
| ENTITY_TYPE | NULLABLE | VARCHAR2 (8) | Indicates whether the business objective is associated with a Program, Project, Proposal, Asset. |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|-----------------------------|------------|----------|-------------|
| RPT_DIM_BUSINESS_OBJ_MV_N15 | NONUNIQUE | 1 | ENTITY_ID |
| RPT_DIM_BUSINESS_OBJ_MV_N16 | NONUNIQUE | 1 | ENTITY_TYPE |

Sequences

This table uses no sequences.

RPT_DIM_FM_BENEFIT_MV

This is a materialized view on top of FM_BENEFITS and FM_USER_DATA tables.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-------------|----------|-----------|-------------------|
| BENEFIT_ID | NOT NULL | NUMBER | ID of the benefit |

| Column Name | Null? | Data Type | Description |
|-----------------------|----------|----------------|--|
| SHOW_ACTUALS_FLAG | NOT NULL | VARCHAR2 (1) | Flag to enable or disable showing of actuals |
| USER_DATA_<N> | NULLABLE | VARCHAR2 (255) | Hidden value of user data associated with the benefit <N> ranges from 1 to 20. |
| USER_DATA_<N>_VISIBLE | NULLABLE | VARCHAR2 (255) | Visible value of user data associated with the benefit <N> ranges from 1 to 20. |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|--------------------------|------------|----------|-------------|
| RPT_DIM_FM_BENEFIT_MV_PK | UNIQUE | 1 | BENEFIT_ID |

Sequences

This table uses no sequences.

RPT_DIM_FM_BENEFIT_LINE_MV

This table stores the individual lines of information per benefit type category over time. The benefit lines have a many-to-one relationship with benefits.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-------------------|----------|---------------|-----------------------------|
| BENEFIT_LINE_ID | NOT NULL | NUMBER | System-generated identifier |
| BENEFIT_TYPE_CODE | NOT NULL | VARCHAR2 (30) | Benefit type code |

| | | | |
|-----------------------|----------|-------------------|--|
| CATEGORY_CODE | NULLABLE | VARCHAR2 (30) | Category associated with the type |
| USER_DATA_<N> | NULLABLE | VARCHAR2 (255) | Hidden value of user data associated with the benefit <N> ranges from 1 to 20. |
| USER_DATA_<N>_VISIBLE | NULLABLE | VARCHAR2 (255) | Visible value of user data associated with the benefit <N> ranges from 1 to 20. |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|-------------------------------|------------|----------|-------------------|
| RPT_DIM_FM_BENEFIT_LINE_MV_PK | UNIQUE | 1 | BENEFIT_LINE_ID |
| RPT_DIM_FM_BENEFIT_LINE_MV_X2 | NONUNIQUE | 1 | BENEFIT_TYPE_CODE |
| RPT_DIM_FM_BENEFIT_LINE_MV_X3 | NONUNIQUE | 1 | CATEGORY_CODE |

Sequences

This table uses no sequences.

RPT_DIM_FM_FCST_ACTUAL_LINE_MV

This is a materialized view on top of PPM's FM_FORECAST_ACTUAL_LINES table.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-------------------------|----------|----------------|--|
| FORECAST_ACTUAL_LINE_ID | NOT NULL | NUMBER | Forecast actual ID for a financial summary derived from FM_FORECAST_ACTUAL_LINES table. |
| EXPENSE_TYPE_CODE | NULLABLE | VARCHAR2 (30) | Indicates the expense type (capital or operating) |
| LABOR_TYPE_CODE | NOT NULL | VARCHAR2 (30) | Labor type (labor or non-labor) |
| CATEGORY_CODE | NOT NULL | VARCHAR2 (30) | Indicates the cost category |
| SYNC_SOURCE_FLAG | NULLABLE | VARCHAR2 (10) | Identifies rolled up lines and lines that are not rolled up. 'N' indicates that the line is not rolled up. 'SP' indicates that the line is rolled up from the staffing profile. 'FS' indicates that the line is rolled up from financial summaries belong to |
| USER_DATA_1-20 | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_1_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_2_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_3_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_4_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_5_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_6_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |

| Column Name | Null? | Data Type | Description |
|----------------------|--------------|------------------|--------------------|
| USER_DATA_7_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_8_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_9_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_10_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_11_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_12_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_13_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_14_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_15_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_16_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_17_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_18_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |

| Column Name | Null? | Data Type | Description |
|----------------------|----------|----------------|-------------------|
| USER_DATA_19_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_20_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|--------------------------------|------------|----------|-------------------------|
| RPT_DIM_FM_FCST_ACTL_LIN_MV_PK | UNIQUE | 1 | FORECAST_ACTUAL_LINE_ID |
| RPT_DIM_FM_FCST_ACTL_LIN_MV_X1 | NONUNIQUE | 1 | EXPENSE_TYPE_CODE |
| RPT_DIM_FM_FCST_ACTL_LIN_MV_X2 | NONUNIQUE | 1 | CATEGORY_CODE |
| RPT_DIM_FM_FCST_ACTL_LIN_MV_X3 | NONUNIQUE | 1 | LABOR_TYPE_CODE |

Sequences

This table uses no sequences.

RPT_DIM_FM_FCST_ACTUAL_MV

This is a materialized view on top of PPM's FM_FORECAST_ACTUALS table.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-------------------------|----------|----------------|---|
| FORECAST_ACTUAL_ID | NOT NULL | NUMBER | ID derived from FM_FORECAST_ACTUALS table. |
| CAPEX_OPEX_ENABLED_FLAG | NOT NULL | VARCHAR2 (1) | Indicates whether the costs and actuals are tracked by capitalization as well. The setting is "Financial Summary tracks capitalized costs". |
| USER_DATA_1-20 | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_1_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_2_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_3_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_4_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_5_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_6_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_7_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_8_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |

| Column Name | Null? | Data Type | Description |
|----------------------|--------------|------------------|--------------------|
| USER_DATA_9_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_10_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_11_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_12_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_13_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_14_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_15_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_16_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_17_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_18_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_19_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |
| USER_DATA_20_VISIBLE | NULLABLE | VARCHAR2 (255) | User data segment |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|----------------------------|------------|----------|--------------------|
| RPT_DIM_FM_FCST_ACTL_MV_PK | UNIQUE | 1 | FORECAST_ACTUAL_ID |

Sequences

This table uses no sequences.

RPT_DIM_FM_FIN_SUMMARY_MV

This is a materialized view on top of PPM's FM_FINANCIAL_SUMMARY table.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|----------------------|----------|-----------------|---|
| FINANCIAL_SUMMARY_ID | NULLABLE | NUMBER | ID of the financial summary derived from FM_FINANCIAL_SUMMARY table |
| NAME | NULLABLE | VARCHAR2 (1300) | Name of the financial summary |
| DESCRIPTION | NULLABLE | VARCHAR2 (1300) | Description of a Financial Summary |
| ENTITY_TYPE | NULLABLE | VARCHAR2 (255) | Identifies the financial summary as the current financial summary (FS) or a snapshot (SNAPSHOT) of a financial summary. |
| CREATED_BY | NULLABLE | VARCHAR2 (200) | Identifier for the user who created the record |

| Column Name | Null? | Data Type | Description |
|-------------------------|----------|---------------|--|
| IS_PLAN_OF_RECORD | NULLABLE | VARCHAR2 (1) | Flag that indicates whether the financial summary is the plan of record (set to 'Y') or is not the plan of record (set to 'N'). |
| LOCAL_CURRENCY_CODE | NULLABLE | VARCHAR2 (40) | Each financial summary has an associated local currency and a code. The local currency and code are based on the region associated with the financial summary's parent. Programs have no associated region, so the local currency defaults to the base currency. |
| BASE_CURRENCY_CODE | NULLABLE | VARCHAR2 (10) | Each financial summary has a base currency and a code associated with it. Example: USD for United States Dollar. The base currency defaults to the PPM Center base currency. |
| SNAPSHOT_SEQ | NULLABLE | NUMBER | Sequence in which snapshots were created, ranging from 0 to n. |
| CREATION_DATE | NULLABLE | DATE | Date record was created |
| IS_FINANCIAL_DATA_TABLE | NULLABLE | VARCHAR2 (1) | Flag that indicates whether the entity ID for the financial summary's parent is set to 19 (Y) or not (N). |
| LOCAL_CURRENCY_SYMBOL | NULLABLE | VARCHAR2 (50) | Symbol for the local currency |
| BASE_CURRENCY_SYMBOL | NULLABLE | VARCHAR2 (50) | Symbol for the base currency |
| FS_ENABLED | NULLABLE | VARCHAR2 (1) | Identifies if the financial summary is enabled 'Y' or not 'N'. |

Indexes

This table uses no indexes.

Sequences

This table uses no sequences.

RPT_DIM_FM_KNTA_LOOKUPS_MV

This is a materialized view on top of PPM's KNTA_LOOKUPS table with additional logic for BUDGET_LABOR_CATEGORY, BUDGET_TYPE, BUDGET_EXPENSE_TYPE lookup types.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-------------------|----------|----------------|---|
| CATEGORY_CODE | NULLABLE | VARCHAR2 (40) | Labor category code or expense type category code |
| CATEGORY_NAME | NULLABLE | VARCHAR2 (150) | Labor category name or expense type category name |
| EXPENSE_TYPE_CODE | NULLABLE | VARCHAR2 (40) | Expense type code of the forecast actual line |
| EXPENSE_TYPE_NAME | NULLABLE | VARCHAR2 (150) | Expense type name of the forecast actual line |
| LABOR_TYPE_CODE | NULLABLE | VARCHAR2 (40) | Labor type code of the forecast actual line |
| LABOR_TYPE_NAME | NULLABLE | VARCHAR2 (150) | Labor type name of the forecast actual line |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|-----------------------------|------------|----------|-------------------|
| RPT_DIM_FM_KNTA_LKUPS_MV_X1 | NONUNIQUE | 1 | EXPENSE_TYPE_CODE |
| RPT_DIM_FM_KNTA_LKUPS_MV_X2 | NONUNIQUE | 1 | CATEGORY_CODE |
| RPT_DIM_FM_KNTA_LKUPS_MV_X3 | NONUNIQUE | 1 | LABOR_TYPE_CODE |

Sequences

This table uses no sequences.

RPT_DIM_ORG_UNITS_MV

This is a materialized view that represents the org unit dimension in Star schema.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|----------------------|----------|-----------------|---|
| ORG_UNIT_ID | NOT NULL | NUMBER | Unique identifier to represent an org unit. It is derived from PPM KRSC_ORG_UNITS table |
| ORG_UNIT_NAME | NOT NULL | VARCHAR2 (200) | Name of the org unit |
| PARENT_ORG_UNIT_ID | NOT NULL | NUMBER | Immediate parent org unit ID |
| PRIMARY_FLAG | NOT NULL | VARCHAR2 (1) | A flag (Y/N) that indicates if the org unit is a primary org unit or not |
| PARENT_ORG_UNIT_NAME | NULLABLE | VARCHAR2 (200) | Name of the immediate parent org unit |
| LEVEL1_ORG_UNIT_ID | NULLABLE | VARCHAR2 (4000) | ID of the first level org unit from the top in the org unit hierarchy |
| LEVEL1_ORG_UNIT_NAME | NULLABLE | VARCHAR2 (4000) | Name of the first level org unit from the top in the org unit hierarchy |
| LEVEL2_ORG_UNIT_ID | NULLABLE | VARCHAR2 (4000) | ID of the second level org unit from the top in the org unit hierarchy |
| LEVEL2_ORG_UNIT_NAME | NULLABLE | VARCHAR2 (4000) | Name of the second level org unit from the top in the org unit hierarchy |

| Column Name | Null? | Data Type | Description |
|-----------------------|----------|-----------------|--|
| LEVEL3_ORG_UNIT_ID | NULLABLE | VARCHAR2 (4000) | ID of the third level org unit from the top in the org unit hierarchy |
| LEVEL3_ORG_UNIT_NAME | NULLABLE | VARCHAR2 (4000) | Name of the third level org unit from the top in the org unit hierarchy |
| LEVEL4_ORG_UNIT_ID | NULLABLE | VARCHAR2 (4000) | ID of the fourth level org unit from the top in the org unit hierarchy |
| LEVEL4_ORG_UNIT_NAME | NULLABLE | VARCHAR2 (4000) | Name of the fourth level org unit from the top in the org unit hierarchy |
| LEVEL5_ORG_UNIT_ID | NULLABLE | VARCHAR2 (4000) | ID of the fifth level org unit from the top in the org unit hierarchy |
| LEVEL5_ORG_UNIT_NAME | NULLABLE | VARCHAR2 (4000) | Name of the fifth level org unit from the top in the org unit hierarchy |
| LEVEL6_ORG_UNIT_ID | NULLABLE | VARCHAR2 (4000) | ID of the sixth level org unit from the top in the org unit hierarchy |
| LEVEL6_ORG_UNIT_NAME | NULLABLE | VARCHAR2 (4000) | Name of the sixth level org unit from the top in the org unit hierarchy |
| LEVEL7_ORG_UNIT_ID | NULLABLE | VARCHAR2 (4000) | ID of the seventh level org unit from the top in the org unit hierarchy |
| LEVEL7_ORG_UNIT_NAME | NULLABLE | VARCHAR2 (4000) | Name of the seventh level org unit from the top in the org unit hierarchy |
| LEVEL8_ORG_UNIT_ID | NULLABLE | VARCHAR2 (4000) | ID of the eighth level org unit from the top in the org unit hierarchy |
| LEVEL8_ORG_UNIT_NAME | NULLABLE | VARCHAR2 (4000) | Name of the eighth level org unit from the top in the org unit hierarchy |
| LEVEL9_ORG_UNIT_ID | NULLABLE | VARCHAR2 (4000) | ID of the ninth level org unit from the top in the org unit hierarchy |
| LEVEL9_ORG_UNIT_NAME | NULLABLE | VARCHAR2 (4000) | Name of the ninth level org unit from the top in the org unit hierarchy |
| LEVEL10_ORG_UNIT_ID | NULLABLE | VARCHAR2 (4000) | ID of the tenth level org unit from the top in the org unit hierarchy |
| LEVEL10_ORG_UNIT_NAME | NULLABLE | VARCHAR2 (4000) | Name of the tenth level org unit from the top in the org unit hierarchy |
| STAFFING_PROFILE_ID | NULLABLE | NUMBER | ID of the staffing profile that this org unit has. This refers to RPT_DIM_RM_STAFF_PROFILES materialized view. |

| Column Name | Null? | Data Type | Description |
|----------------------------|----------|-----------------|---|
| ORG_UNIT_MANAGER_ID | NULLABLE | VARCHAR2 (40) | User ID of the manager of the org unit |
| ORG_UNIT_MANAGER_FIRSTNAME | NULLABLE | VARCHAR2 (80) | First Name of the manager of the org unit |
| ORG_UNIT_MANAGER_LASTNAME | NULLABLE | VARCHAR2 (80) | Last name of the manager of the org unit |
| FINANCIAL_SUMMARY_ID | NULLABLE | NUMBER | ID of the org unit's financial summary |
| FINANCIAL_SUMMARY_NAME | NOT NULL | VARCHAR2 (1300) | Name of the org unit's financial summary name |
| LOCATION | NULLABLE | VARCHAR2 (150) | Location of the org unit |
| DEPARTMENT | NULLABLE | VARCHAR2 (150) | Department this org unit belongs to |
| CATEGORY | NULLABLE | VARCHAR2 (150) | Category of this org unit |
| SOURCE | NULLABLE | VARCHAR2 (100) | Source of the org unit |
| SOURCE_TYPE_CODE | NULLABLE | VARCHAR2 (30) | Code of the source of the org unit |
| CREATED_BY | NOT NULL | NUMBER | Identifier for the user who created the record |
| REGION_ID | NULLABLE | NUMBER | ID of the region that the org unit belongs to |
| REGION_NAME | NULLABLE | VARCHAR2 (100) | Name of the region that the org unit belongs to |
| USER_DATA_SET_CONTEXT_ID | NULLABLE | NUMBER | Parameter set context identifier for the user data fields |
| USER_DATA1-20 | NULLABLE | VARCHAR2 (200) | User data segment |
| VISIBLE_USER_DATA1-20 | NULLABLE | VARCHAR2 (200) | User data segment |

Indexes

This table uses no indexes.

Sequences

This table uses no sequences.

RPT_DIM_PM_DTL_CSTPARMS_IRS_MV

This is a materialized view that holds the detail custom parameters for project's issue, risk and scope change entities.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-------------------|----------|----------------|--|
| REQUEST_ID | NULLABLE | NUMBER | ID of the request of this issue, risk or scope change entity |
| REQUEST_TYPE_ID | NULLABLE | NUMBER | Request type ID for issue, risk or scope change |
| BATCH_NUMBER | NULLABLE | NUMBER | This column is used when the custom parameters are more than 50 and the 50 columns provided in the table is insufficient. This corresponds to the PPM table's batch number. For example, if the BATCH_NUMBER is 2 and the parameter number is 4, it is considered as parameter 54. |
| REQUEST_TYPE_NAME | NULLABLE | VARCHAR2 (80) | Name of the request type for issue, risk or scope change request |
| PARAMETER1-40 | NULLABLE | VARCHAR2 (200) | Value of custom detail parameters used by the customer for issue, risk or scope change request |

| Column Name | Null? | Data Type | Description |
|------------------------|----------|-----------------|---|
| VISIBLE_PARAMETER1-40 | NULLABLE | VARCHAR2 (200) | Visible value of custom detail parameters used by the customer for issue, risk or scope change request. |
| PARAMETER41-50 | NULLABLE | VARCHAR2 (4000) | Value of custom detail parameters used by the customer for issue, risk or scope change request. |
| VISIBLE_PARAMETER41-50 | NULLABLE | VARCHAR2 (4000) | Visible value of custom detail parameters used by the customer for issue, risk or scope change request. |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|--------------------------------|------------|----------|-----------------|
| RPT_PM_DTL_CSTPARMS_IRS_MV_N14 | NONUNIQUE | 1 | REQUEST_ID |
| RPT_PM_DTL_CSTPARMS_IRS_MV_N15 | NONUNIQUE | 1 | REQUEST_TYPE_ID |

Sequences

This table uses no sequences.

RPT_DIM_PM_HDR_CSTPARMS_IRS_MV

This is a materialized view that holds the header custom parameters for project's issue, risk and scope change entities.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-------------|----------|-----------|---|
| REQUEST_ID | NULLABLE | NUMBER | Request ID for this issue or risk or scope change request |

| Column Name | Null? | Data Type | Description |
|------------------------|----------|-----------------|--|
| REQUEST_TYPE_ID | NULLABLE | NUMBER | Request type ID for issue, risk or scope change |
| BATCH_NUMBER | NULLABLE | NUMBER | This column is used when the custom parameters are more than 50 and the 50 columns provided in the table is insufficient. This corresponds to the PPM table's batch number. For example, if the BATCH_NUMBER is 2 and the parameter number is 4, it is considered as parameter 54. |
| REQUEST_TYPE_NAME | NULLABLE | VARCHAR2 (80) | Name of the request type for issue, risk or scope change request |
| PARAMETER1-40 | NULLABLE | VARCHAR2 (200) | Value of custom header parameters used by the customer for issue, risk or scope change request |
| VISIBLE_PARAMETER1-40 | NULLABLE | VARCHAR2 (200) | Visible value of custom header parameters used by the customer for issue, risk or scope change request. |
| PARAMETER41-50 | NULLABLE | VARCHAR2 (4000) | Value of custom header parameters used by the customer for issue, risk or scope change request. |
| VISIBLE_PARAMETER41-50 | NULLABLE | VARCHAR2 (4000) | Visible value of custom header parameters used by the customer for issue, risk or scope change request. |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|--------------------------------|------------|----------|-----------------|
| RPT_PM_HDR_CSTPARMS_IRS_MV_N12 | NONUNIQUE | 1 | REQUEST_ID |
| RPT_PM_HDR_CSTPARMS_IRS_MV_N13 | NONUNIQUE | 1 | REQUEST_TYPE_ID |

Sequences

This table uses no sequences.

RPT_DIM_PM_PROJ_ISSUE_MV

This is a materialized view that holds the information of the project issues.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|---------------------------|----------|--------------------|--|
| PROJECT_ID | NULLABLE | NUMBER | ID that identifies the project |
| REQUEST_ID | NOT NULL | NUMBER | Issue Request ID associated with this project |
| DESCRIPTION | NULLABLE | VARCHAR2 (400) | Description of this issue |
| REQUEST_TYPE_ID | NOT NULL | NUMBER | The request type ID for this issue request |
| REQUEST_TYPE_NAME | NOT NULL | VARCHAR2 (80) | The request type name for this issue request |
| ENTITY_STATUS | NOT NULL | VARCHAR2 (80) | Status (lookup value) of this issue |
| PRIORITY | NULLABLE | VARCHAR2 (150) | Priority (lookup value) of this issue |
| DEPARTMENT | NULLABLE | VARCHAR2 (150) | Department (lookup value) of this issue |
| APPLICATION | NULLABLE | VARCHAR2 (150) | Application (lookup value) of this issue |
| WORKFLOW_NAME | NOT NULL | VARCHAR2 (150) | This is the request workflow name |
| ACTIVE_WORKFLOW_STEP_NAME | NULLABLE | VARCHAR2 (4000) | Name of the current active step(s) available in the workflow associated with the Issue request |
| CREATED_BY | NULLABLE | VARCHAR2 (200) | Identifier for the user who created the record |
| CREATION_DATE | NOT NULL | DATE | Date record was created |
| CONTACT_NAME | NULLABLE | VARCHAR2 (200) | Full name of the contact person of this issue |

| Column Name | Null? | Data Type | Description |
|--------------------------|----------|-------------------|---|
| CONTACT_PHONE | NULLABLE | VARCHAR2 (30) | Phone number of the contact person of this issue |
| CONTACT_EMAIL | NULLABLE | VARCHAR2 (80) | Email address of the contact person of this issue |
| ASSIGNED_TO | NULLABLE | VARCHAR2 (200) | The person to whom this issue is assigned to |
| USER_DATA1-20 | NULLABLE | VARCHAR2 (200) | User data segment |
| VISIBLE_USER_DATA1-20 | NULLABLE | VARCHAR2 (200) | User data segment |
| ESCALATION_LEVEL_MEANING | NULLABLE | VARCHAR2 (200) | Escalation level, this information is stored at Issue Field Group table |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|-----------------------------|------------|----------|-----------------|
| RPT_DIM_PM_PROJ_ISSUE_MV_N1 | NONUNIQUE | 1 | PROJECT_ID |
| RPT_DIM_PM_PROJ_ISSUE_MV_N2 | NONUNIQUE | 1 | REQUEST_ID |
| RPT_DIM_PM_PROJ_ISSUE_MV_N3 | NONUNIQUE | 1 | REQUEST_TYPE_ID |

Sequences

This table uses no sequences.

RPT_DIM_PM_PROJ_RISK_MV

This is a materialized view that holds the information of the project risks.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|---------------------------|----------|--------------------|---|
| PROJECT_ID | NULLABLE | NUMBER | ID that identifies the project |
| REQUEST_ID | NOT NULL | NUMBER | Risk Request ID associated with this project |
| DESCRIPTION | NULLABLE | VARCHAR2 (400) | Description of this risk |
| REQUEST_TYPE_ID | NOT NULL | NUMBER | The request type ID for this risk request |
| REQUEST_TYPE_NAME | NOT NULL | VARCHAR2 (80) | The request type name for this risk request |
| ENTITY_STATUS | NOT NULL | VARCHAR2 (80) | Status (lookup value) of this risk |
| PRIORITY | NULLABLE | VARCHAR2 (150) | Priority (lookup value) of this risk |
| DEPARTMENT | NULLABLE | VARCHAR2 (150) | Department (lookup value) of this risk |
| APPLICATION | NULLABLE | VARCHAR2 (150) | Application (lookup value) of this risk |
| WORKFLOW_NAME | NOT NULL | VARCHAR2 (150) | This is the request workflow name |
| ACTIVE_WORKFLOW_STEP_NAME | NULLABLE | VARCHAR2 (4000) | Name of the current active step(s) available in the workflow associated with the Risk request |
| CREATED_BY | NULLABLE | VARCHAR2 (200) | Identifier for the user who created the record |
| CREATION_DATE | NOT NULL | DATE | Date record was created |
| CONTACT_NAME | NULLABLE | VARCHAR2 (200) | Full name of the contact person of this risk |
| CONTACT_PHONE | NULLABLE | VARCHAR2 (30) | Phone number of the contact person of this risk |
| CONTACT_EMAIL | NULLABLE | VARCHAR2 (80) | Email address of the contact person of this risk |

| Column Name | Null? | Data Type | Description |
|---------------------------|----------|-------------------|---|
| ASSIGNED_TO | NULLABLE | VARCHAR2 (200) | The person to whom this risk is assigned to |
| USER_DATA1-20 | NULLABLE | VARCHAR2 (200) | User data segment |
| VISIBLE_USER_DATA1-20 | NULLABLE | VARCHAR2 (200) | User data segment |
| PROBABILITY_MEANING | NULLABLE | VARCHAR2 (200) | Probability (lookup value) of this risk |
| RISK_IMPACT_LEVEL_MEANING | NULLABLE | VARCHAR2 (200) | Impact level (lookup value) of this risk |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|----------------------------|------------|----------|-----------------|
| RPT_DIM_PM_PROJ_RISK_MV_N7 | NONUNIQUE | 1 | PROJECT_ID |
| RPT_DIM_PM_PROJ_RISK_MV_N8 | NONUNIQUE | 1 | REQUEST_ID |
| RPT_DIM_PM_PROJ_RISK_MV_N9 | NONUNIQUE | 1 | REQUEST_TYPE_ID |

Sequences

This table uses no sequences.

RPT_DIM_PM_PROJ_SCPCHG_MV

This is a materialized view that holds the information of the project Scope Change.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|---------------------------|----------|-----------------|---|
| PROJECT_ID | NULLABLE | NUMBER | ID that identifies the project |
| REQUEST_ID | NOT NULL | NUMBER | Scope Change Request ID associated with this project |
| REQUEST_TYPE_ID | NOT NULL | NUMBER | The request type ID for this scope change request |
| REQUEST_TYPE_NAME | NOT NULL | VARCHAR2 (80) | The request type name for this scope change request |
| DESCRIPTION | NULLABLE | VARCHAR2 (400) | Description of this scope change |
| ENTITY_STATUS | NOT NULL | VARCHAR2 (80) | Status (lookup value) of this scope change |
| PRIORITY | NULLABLE | VARCHAR2 (150) | Priority (lookup value) of this scope change |
| DEPARTMENT | NULLABLE | VARCHAR2 (150) | Department (lookup value) of this scope change |
| APPLICATION | NULLABLE | VARCHAR2 (150) | Application (lookup value) of this scope change |
| WORKFLOW_NAME | NOT NULL | VARCHAR2 (150) | This is the request workflow name |
| ACTIVE_WORKFLOW_STEP_NAME | NULLABLE | VARCHAR2 (4000) | Name of the current active step(s) available in the workflow associated with the Scope Change request |
| CREATED_BY | NULLABLE | VARCHAR2 (200) | Identifier for the user who created the record |
| CREATION_DATE | NOT NULL | DATE | Date record was created |
| CONTACT_NAME | NULLABLE | VARCHAR2 (200) | Full name of the contact person of this scope change |
| CONTACT_PHONE | NULLABLE | VARCHAR2 (30) | Phone number of the contact person of this scope change |
| CONTACT_EMAIL | NULLABLE | VARCHAR2 (80) | Email address of the contact person of this scope change |

| Column Name | Null? | Data Type | Description |
|-------------------------|----------|-------------------|---|
| ASSIGNED_TO | NULLABLE | VARCHAR2 (200) | The person to whom this scope change is assigned to |
| USER_DATA1-20 | NULLABLE | VARCHAR2 (200) | User data segment |
| VISIBLE_USER_DATA1-20 | NULLABLE | VARCHAR2 (200) | User data segment |
| IMPACT_SEVERITY_MEANING | NULLABLE | VARCHAR2 (200) | Impact severity (lookup value) of this scope change |
| CR_LEVEL_MEANING | NULLABLE | VARCHAR2 (200) | CR Level (lookup value) of this scope change |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|------------------------------|------------|----------|-----------------|
| RPT_DIM_PM_PROJ_SCPCHG_MV_N4 | NONUNIQUE | 1 | PROJECT_ID |
| RPT_DIM_PM_PROJ_SCPCHG_MV_N5 | NONUNIQUE | 1 | REQUEST_ID |
| RPT_DIM_PM_PROJ_SCPCHG_MV_N6 | NONUNIQUE | 1 | REQUEST_TYPE_ID |

Sequences

This table uses no sequences.

RPT_DIM_PM_PROJ_SETTINGS_MV

This is a materialized view representing the Project Setting dimension in the Star schema.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-------------|----------|-----------|---|
| PROJECT_ID | NOT NULL | NUMBER | Unique identifier of a project derived from PPM PM_PROJECTS table |
| CE_ERE | NULLABLE | NUMBER | It provides Estimated Remaining Effort setting for the project. Value is 1 if checkbox is checked in PPM, otherwise 0 |
| CE_USETM | NULLABLE | NUMBER | It provides Use TM setting for the project. Value is 1 if checkbox is checked in PPM, otherwise 0 |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|-----------------------------|------------|----------|---------------|
| I_SNAP\$_RPT_DIM_PM_PROJ_SE | UNIQUE | 1 | SYS_NC00004\$ |

Sequences

This table uses no sequences.

RPT_DIM_PM_TASKS

This table holds the task information that is derived from PPM table. All the fields are derived from PPM task table.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-----------------------|----------|-------------------|--|
| TASK_ID | NOT NULL | NUMBER | Unique ID of the TASK table |
| PROJECT_ID | NULLABLE | NUMBER | Identifier of the related project |
| WORK_PLAN_ID | NULLABLE | NUMBER | Identifier of the related work plan |
| TASK_NAME | NULLABLE | VARCHAR2 (300) | Name of the task |
| TASK_DESCRIPTION | NULLABLE | VARCHAR2 (300) | Description of the task |
| PRIORITY | NULLABLE | NUMBER | Priority of the task |
| STATUS | NULLABLE | VARCHAR2 (150) | Status of the task |
| TASK_TYPE | NULLABLE | VARCHAR2(1) | Type of the task |
| MILESTONE | NULLABLE | VARCHAR2(1) | It indicates if task is milestone |
| MAJOR_MILESTONE | NULLABLE | VARCHAR2(1) | It indicates if task is major milestone |
| IN_CRITICAL_PATH | NULLABLE | VARCHAR2(1) | It indicates if task is in critical path |
| PARENT_TASK_ID | NULLABLE | NUMBER | ID of parent task |
| PARENT_TASK_NAME | NULLABLE | VARCHAR2 (300) | Name of parent task |
| SCHED_START_DATE | NULLABLE | DATE | Schedule start date of the task |
| SCHED_FINISH_DATE | NULLABLE | DATE | Schedule finish date of the task |
| SCHED_DURATION | NULLABLE | FLOAT | Schedule duration of the task |
| PERC_COMPLETE | NULLABLE | FLOAT | Percent Complete of the task |
| ACTUAL_START_DATE | NULLABLE | DATE | Actual start date of the task |
| ACTUAL_FINISH_DATE | NULLABLE | DATE | Actual finish date of the task |
| ACTUAL_DURATION | NULLABLE | FLOAT | Actual duration of the task |
| ESTIMATED_FINISH_DATE | NULLABLE | DATE | Estimated finish date of the task |

| Column Name | Null? | Data Type | Description |
|-----------------------|----------|----------------|---|
| CREATED_BY_FIRSTNAME | NULLABLE | VARCHAR2 (80) | First Name of user who creates the task |
| CREATED_BY_LASTNAME | NULLABLE | VARCHAR2 (80) | Last Name of user who creates the task |
| CREATION_DATE | NULLABLE | DATE | Task creation date |
| SERVICE_ID | NULLABLE | VARCHAR2 (40) | ID of the related service |
| SERVICE_NAME | NULLABLE | VARCHAR2 (255) | Name of the related service |
| SCHEDULE_HEALTH | NULLABLE | VARCHAR2 (255) | Schedule health of the task |
| EXCEPTION_TASK_COUNT | NULLABLE | NUMBER | Number of exception in task |
| TOTAL_EXCEPTION_COUNT | NULLABLE | NUMBER | Total exception count in the project |
| DATUM1_CODE | NULLABLE | VARCHAR2 (255) | User data code 1 |
| DATUM1_VISIBLE | NULLABLE | VARCHAR2 (255) | Visible value of user data code 1 |
| DATUM2_CODE | NULLABLE | VARCHAR2 (255) | User data code 2 |
| DATUM2_VISIBLE | NULLABLE | VARCHAR2 (255) | Visible value of user data code 2 |
| DATUM3_CODE | NULLABLE | VARCHAR2 (255) | User data code 3 |
| DATUM3_VISIBLE | NULLABLE | VARCHAR2 (255) | Visible value of user data code 3 |
| DATUM4_CODE | NULLABLE | VARCHAR2 (255) | User data code 4 |
| DATUM4_VISIBLE | NULLABLE | VARCHAR2 (255) | Visible value of user data code 4 |
| DATUM5_CODE | NULLABLE | VARCHAR2 (255) | User data code 5 |

| Column Name | Null? | Data Type | Description |
|-----------------|----------|-------------------|------------------------------------|
| DATUM5_VISIBLE | NULLABLE | VARCHAR2 (255) | Visible value of user data code 5 |
| DATUM6_CODE | NULLABLE | VARCHAR2 (255) | User data code 6 |
| DATUM6_VISIBLE | NULLABLE | VARCHAR2 (255) | Visible value of user data code 6 |
| DATUM7_CODE | NULLABLE | VARCHAR2 (255) | User data code 7 |
| DATUM7_VISIBLE | NULLABLE | VARCHAR2 (255) | Visible value of user data code 7 |
| DATUM8_CODE | NULLABLE | VARCHAR2 (255) | User data code 8 |
| DATUM8_VISIBLE | NULLABLE | VARCHAR2 (255) | Visible value of user data code 8 |
| DATUM9_CODE | NULLABLE | VARCHAR2 (255) | User data code 9 |
| DATUM9_VISIBLE | NULLABLE | VARCHAR2 (255) | Visible value of user data code 9 |
| DATUM10_CODE | NULLABLE | VARCHAR2 (255) | User data code 10 |
| DATUM10_VISIBLE | NULLABLE | VARCHAR2 (255) | Visible value of user data code 10 |
| DATUM11_CODE | NULLABLE | VARCHAR2 (255) | User data code 11 |
| DATUM11_VISIBLE | NULLABLE | VARCHAR2 (255) | Visible value of user data code 11 |
| DATUM12_CODE | NULLABLE | VARCHAR2 (255) | User data code 12 |
| DATUM12_VISIBLE | NULLABLE | VARCHAR2 (255) | Visible value of user data code 12 |
| DATUM13_CODE | NULLABLE | VARCHAR2 (255) | User data code 13 |
| DATUM13_VISIBLE | NULLABLE | VARCHAR2 (255) | Visible value of user data code 13 |

| Column Name | Null? | Data Type | Description |
|-----------------|----------|-------------------|------------------------------------|
| DATUM14_CODE | NULLABLE | VARCHAR2 (255) | User data code 14 |
| DATUM14_VISIBLE | NULLABLE | VARCHAR2 (255) | Visible value of user data code 14 |
| DATUM15_CODE | NULLABLE | VARCHAR2 (255) | User data code 15 |
| DATUM15_VISIBLE | NULLABLE | VARCHAR2 (255) | Visible value of user data code 15 |
| DATUM16_CODE | NULLABLE | VARCHAR2 (255) | User data code 16 |
| DATUM16_VISIBLE | NULLABLE | VARCHAR2 (255) | Visible value of user data code 16 |
| DATUM17_CODE | NULLABLE | VARCHAR2 (255) | User data code 17 |
| DATUM17_VISIBLE | NULLABLE | VARCHAR2 (255) | Visible value of user data code 17 |
| DATUM18_CODE | NULLABLE | VARCHAR2 (255) | User data code 18 |
| DATUM18_VISIBLE | NULLABLE | VARCHAR2 (255) | Visible value of user data code 18 |
| DATUM19_CODE | NULLABLE | VARCHAR2 (255) | User data code 19 |
| DATUM19_VISIBLE | NULLABLE | VARCHAR2 (255) | Visible value of user data code 19 |
| DATUM20_CODE | NULLABLE | VARCHAR2 (255) | User data code 20 |
| DATUM20_VISIBLE | NULLABLE | VARCHAR2 (255) | Visible value of user data code 20 |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|---------------------|------------|----------|-------------|
| RPT_DIM_PM_TASKS_N1 | NONUNIQUE | 1 | TASK_ID |

| Index Name | Index Type | Sequence | Column Name |
|---------------------|------------|----------|----------------|
| RPT_DIM_PM_TASKS_N2 | NONUNIQUE | 1 | PROJECT_ID |
| RPT_DIM_PM_TASKS_N3 | NONUNIQUE | 1 | PARENT_TASK_ID |
| RPT_DIM_PM_TASKS_N4 | NONUNIQUE | 1 | STATUS |

Sequences

This table uses no sequences.

RPT_DIM_PM_WORKPLAN_MV

This is a materialized view representing work plan dimension in the Star schema.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|----------------------|----------|----------------|---|
| WORK_PLAN_ID | NOT NULL | NUMBER | Unique identifier of a work plan derived from PPM PM_PROJECTS table |
| PROJECT_ID | NULLABLE | NUMBER | ID of the related project. It is derived from pm_projects table |
| ROOT_TASK_ID | NULLABLE | NUMBER | ID of the root task in the work plan |
| WORKPLAN_NAME | NULLABLE | VARCHAR2 (300) | Name of the work plan |
| DESCRIPTION | NULLABLE | VARCHAR2 (650) | Description of the work plan. |
| CREATED_BY_FIRSTNAME | NOT NULL | VARCHAR2 (80) | First name of the user who creates this work plan |

| Column Name | Null? | Data Type | Description |
|-----------------------------|----------|------------------|---|
| CREATED_ BY_ LASTNAME | NOT NULL | VARCHAR2 (80) | Last name of the user who creates this work plan |
| CREATION_ DATE | NOT NULL | DATE | Work plan creation date |
| IS_BASELINE_ FLAG | NULLABLE | CHAR | A flag to indicate if a workplan is a baseline workplan. Value of 'Y' indicates it is a baseline workplan. |
| WORKPLAN_ SEQ | NULLABLE | NUMBER | Sequence number of the work plan |
| IS_ACTIVE_ FLAG | NULLABLE | VARCHAR2 (1) | A flag to indicate if a workplan is an active baseline work plan. Value of 'Y' indicates it is an active baseline workplan. |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|----------------------------|------------|----------|--------------|
| RPT_DIM_PM_WORKPLAN_MV_N10 | NONUNIQUE | 1 | WORK_PLAN_ID |
| RPT_DIM_PM_WORKPLAN_MV_N11 | NONUNIQUE | 1 | PROJECT_ID |

Sequences

This table uses no sequences.

RPT_DIM_PORTFOLIOS

This is a materialized view on top of PPM's PFM_PORTFOLIOS table.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-----------------------|----------|-----------------|--|
| PORTFOLIO_ID | NOT NULL | NUMBER | ID of the portfolio |
| NAME | NOT NULL | VARCHAR2 (200) | Name of the portfolio |
| LEVEL_1_PARENT_ID | NULLABLE | VARCHAR2 (4000) | ID of the top level parent portfolio of this portfolio |
| LEVEL_1_PARENT_NAME | NULLABLE | VARCHAR2 (4000) | Name of the top level parent portfolio of this portfolio |
| LEVEL_<N>_PARENT_ID | NULLABLE | VARCHAR2 (4000) | ID of the Nth level parent portfolio of this portfolio |
| LEVEL_<N>_PARENT_NAME | NULLABLE | VARCHAR2 (4000) | Name of the Nth level parent portfolio of this portfolio |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|------------------------|------------|----------|--------------|
| RPT_DIM_PORTFOLIOS_N10 | NONUNIQUE | 1 | PORTFOLIO_ID |
| RPT_DIM_PORTFOLIOS_N11 | NONUNIQUE | 1 | NAME |

Sequences

This table uses no sequences.

RPT_DIM_PORTFOLIO MANAGERS_MV

This is a materialized view on top of PFM_POTFOLIO MANAGERS, RSC_RESOURCES, and KNTA_USERS tables.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-------------------|----------|-----------------|--|
| PORTFOLIO_ID | NOTNULL | NUMBER | ID of the portfolio |
| USER_ID | NOTNULL | NUMBER | ID of the portfolio manager |
| RESOURCE_ID | NULLABLE | NUMBER | ID of the resource derived from PPM RSC_RESOURCES table |
| USERNAME | NULLABLE | VARCHAR2 (200) | Portfolio manager's username derived from KNTA_USERS table |
| FIRST_NAME | NULLABLE | VARCHAR2 (80) | Portfolio manager's first name derived from KNTA_USERS table |
| LAST_NAME | NULLABLE | VARCHAR2 (80) | Portfolio manager's last name derived from KNTA_USERS table |
| FULL_NAME | NULLABLE | VARCHAR2 (200) | Portfolio manager's full name derived from KNTA_USERS table |
| ALL_PORTFOLIO_MGR | NULLABLE | VARCHAR2 (3999) | Full names of all managers of the portfolio. The names are separated by comma. |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|---------------------------|------------|----------|--------------|
| PORTFOLIO MANAGERS_MV_N12 | NONUNIQUE | 1 | PORTFOLIO_ID |
| PORTFOLIO MANAGERS_MV_N13 | NONUNIQUE | 1 | USER_ID |

Sequences

This table uses no sequences.

RPT_DIM_PPM_FISCAL_PERIODS

This is a dimensional table in the reporting star schema. It is derived from the PPM_FISCAL_PERIODS table. All the period names (week, month, quarter, half year and year) are stored at day level.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-----------------------|----------|----------------|--|
| DATE_OF_THE_DAY | NULLABLE | DATE | Date in the calendar that is between start and the end etl boundaries. |
| PERIOD_ID_WEEK | NULLABLE | NUMBER | ID of the (parent) week in which this date falls in. |
| PERIOD_ID_MONTH | NULLABLE | NUMBER | ID of the (parent) month in which this date falls in. |
| PERIOD_ID_QUARTER | NULLABLE | NUMBER | ID of the (parent) quarter in which this date falls in. |
| PERIOD_ID_HALF_YEAR | NULLABLE | NUMBER | ID of the (parent) half year in which this date falls in. |
| PERIOD_ID_YEAR | NULLABLE | NUMBER | ID of the (parent) year in which this date falls in. |
| PERIOD_NAME_WEEK | NULLABLE | VARCHAR2 (100) | Name of the (parent) week in which this date falls in. |
| PERIOD_NAME_MONTH | NULLABLE | VARCHAR2 (100) | Name of the (parent) month in which this date falls in. |
| PERIOD_NAME_QUARTER | NULLABLE | VARCHAR2 (100) | Name of the (parent) quarter in which this date falls in. |
| PERIOD_NAME_HALF_YEAR | NULLABLE | VARCHAR2 (100) | Name of the (parent) half year in which this date falls in. |

| Column Name | Null? | Data Type | Description |
|--------------------|----------|----------------|--|
| PERIOD_NAME_YEAR | NULLABLE | VARCHAR2 (100) | Name of the (parent) year in which this date falls in. |
| BEGIN_FISCAL_MONTH | NULLABLE | CHAR | A flag to indicate if this date is the beginning of the month. |
| BEGIN_FISCAL_YEAR | NULLABLE | CHAR | A flag to indicate if this date is the beginning of the year. |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|-------------------------------|------------|----------|--------------------|
| RPT_DIM_PPM_FISCAL_PERIODS_N1 | NONUNIQUE | 1 | PERIOD_NAME_MONTH |
| RPT_DIM_PPM_FISCAL_PERIODS_N2 | NONUNIQUE | 1 | PERIOD_ID_MONTH |
| RPT_DIM_PPM_FISCAL_PERIODS_N3 | NONUNIQUE | 1 | BEGIN_FISCAL_MONTH |
| RPT_DIM_PPM_FISCAL_PERIODS_N4 | NONUNIQUE | 1 | DATE_OF_THE_DAY |
| RPT_DIM_PPM_FISCAL_PERIODS_N5 | NONUNIQUE | 1 | PERIOD_ID_YEAR |
| RPT_DIM_PPM_FISCAL_PERIODS_N6 | NONUNIQUE | 1 | BEGIN_FISCAL_YEAR |
| RPT_DIM_PPM_FISCAL_PERIODS_N7 | NONUNIQUE | 1 | PERIOD_NAME_WEEK |

Sequences

This table uses no sequences.

RPT_DIM_PROGRAM_CONTENT_MV

This is a materialized view on top of PPM's PGM_PROGRAM_CONTENT table.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|---------------|----------|---------------|---|
| PROGRAM_ID | NOT NULL | NUMBER | ID of a program derived from PGM_PROGRAM_CONTENT table. |
| LIFECYCLE_ID | NOT NULL | NUMBER | Identifier for the parent of the financial summary. Based on PARENT_ENTITY_ID, the PARENT_ID could be LIFECYCLE_ID in PFM_LIFECYCLE_PARENT_ENTITY, ORG_UNIT_ID in KRSC_ORG_UNITS_NLS, PROGRAM_ID in PGM_PROGRAMS or REQUEST_ID in KCRT_REQUESTS |
| ACTIVE_REQ_ID | NULLABLE | NUMBER | Identifier of the active request Id |
| ACTIVE_ENTITY | NOT NULL | VARCHAR2 (20) | Active entity name of the program |

Indexes

This table uses no indexes.

Sequences

This table uses no sequences.

RPT_DIM_PROGRAMS_MV

This is a materialized view on top of PPM's PGM_PROGRAMS, PFM_PORTFOLIOS, and PFM_PORTFOLIO_CONTENTS tables.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-----------------------|----------|----------------|---|
| PROGRAM_ID | NOT NULL | NUMBER | ID of the program derived from PGM_PROGRAMS table |
| PROGRAM_NAME | NULLABLE | VARCHAR2 (255) | Name of the program |
| FINANCIAL_SUMMARY_ID | NULLABLE | NUMBER | ID of a financial summary associated with this program derived from PGM_PROGRAMS table. |
| PORTFOLIO_ID | NULLABLE | NUMBER | ID of the portfolio this program belongs to |
| PORTFOLIO_NAME | NULLABLE | VARCHAR2 (200) | Name of the portfolio this program belongs to |
| USER_DATA1-20 | NULLABLE | VARCHAR2 (255) | User data segment |
| VISIBLE_USER_DATA1-20 | NULLABLE | VARCHAR2 (255) | User data segment |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|------------------------|------------|----------|--------------|
| PKC_PGM_PROGRAMS1 | UNIQUE | 1 | PROGRAM_ID |
| RPT_DIM_PROGRAMS_MV_N5 | NONUNIQUE | 1 | PROGRAM_NAME |
| RPT_DIM_PROGRAMS_MV_N3 | NONUNIQUE | 1 | PORTFOLIO_ID |

Sequences

This table uses no sequences.

RPT_DIM_PROJECTS_MV

This is a materialized view representing the Project dimension in the Star schema.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|------------------------|----------|----------------------|--|
| PROJECT_ID | NULLABLE | NUMBER | Unique identifier of a project derived from PPM PM_PROJECTS table |
| REQUEST_ID | NULLABLE | NUMBER | ID of the request associated with the project |
| PROJECT_NAME | NULLABLE | VARCHAR2 (300) | Name of the project |
| PROJECT_DESCRIPTION | NULLABLE | VARCHAR2 (650) | Description of the project |
| DIRECT_PORTFOLIO_ID | NULLABLE | NUMBER | ID of the portfolio this project directly belongs to |
| DIRECT_PORTFOLIO_NAME | NULLABLE | VARCHAR2 (1500 CHAR) | Name of the portfolio this project directly belongs to |
| REGION_ID | NULLABLE | NUMBER | ID of the associated region with the project |
| ASSOCIATED_MSP_PROJECT | NULLABLE | VARCHAR2 (255) | The file name and path of the microsoft project if the project is integrated with any MSP file |
| PROJECT_MANAGER | NULLABLE | VARCHAR2 (4000) | Name of the managers of the project. Names are separated by comma for two or more project managers for the project |
| START_DATE_PERIOD | NULLABLE | VARCHAR2 (100) | Start period name of the project |

| Column Name | Null? | Data Type | Description |
|------------------------|----------|----------------|--|
| FINISH_DATE_PERIOD | NULLABLE | VARCHAR2 (100) | Finish period name of the project |
| START_DATE_PERIOD_ID | NULLABLE | NUMBER | ID of the Start period of the project |
| FINISH_DATE_PERIOD_ID | NULLABLE | NUMBER | ID of the end period of the project |
| PROJECT_TYPE_ID | NULLABLE | NUMBER | ID of the Project type this project is created from. It is derived from PROJECT_TYPE_ID of PPM PM_PROJECT_TYPES table. |
| PROJECT_DISPLAY_STATUS | NULLABLE | VARCHAR2 (255) | Status of the project |
| PROJECT_STATUS | NULLABLE | NUMBER | Internal status code of the project |
| PROJECT_ACCESS_ID | NULLABLE | NUMBER | Provides project access control information. It is derived from SECURITY_ENTITY_ID of PPM ITG_SECURABLE_ENTITIES table. |
| PROJECT_COST_ACCESS_ID | NULLABLE | NUMBER | Provides project cost access control information. It is derived from SECURITY_ENTITY_ID of ITG_SECURABLE_ENTITIES table. |
| PROJECT_CREATED_BY | NULLABLE | VARCHAR2 (200) | ID of the user who created the project |
| PROJECT_CREATION_DATE | NULLABLE | DATE | Project creation date |
| ACTUAL_START_DATE | NULLABLE | DATE | Actual start date of the project |
| ACTUAL_FINISH_DATE | NULLABLE | DATE | Actual finish date of the project |
| LAST_UPDATE_DATE | NULLABLE | DATE | Date record was last updated |
| PROJECT_TYPE_NAME | NULLABLE | VARCHAR2 (255) | Name of the project type |

| Column Name | Null? | Data Type | Description |
|---------------------------|----------|----------------|---|
| SCHEDULE_HEALTH_INDICATOR | NULLABLE | VARCHAR2 (255) | A flag for schedule health of the project |
| ISSUE_HEALTH_INDICATOR | NULLABLE | VARCHAR2 (255) | A flag for issue health of the project |
| COST_HEALTH_INDICATOR | NULLABLE | VARCHAR2 (255) | A flag for cost health of the project |
| OVERALL_HEALTH_INDICATOR | NULLABLE | VARCHAR2 (255) | A flag for overall health of the project |
| PERCENT_COMPLETE | NULLABLE | FLOAT | Percent complete |
| ACT_DURATION | NULLABLE | FLOAT | Actual duration of the project |
| SCHED_DURATION | NULLABLE | FLOAT | Schedule duration of the project |
| EST_FINISH_DATE | NULLABLE | DATE | Estimated finish date of the project |
| SCHED_START_DATE | NULLABLE | DATE | Scheduled start date of the project |
| SCHED_FINISH_DATE | NULLABLE | DATE | Scheduled finish date of the project |

Indexes

This table uses no indexes.

Sequences

This table uses no sequences.

RPT_DIM_REGIONS_MV

This is a materialized view over PPM's KNTA_REGIONS, KDRV_CALENDARS and KCST_CURRENCIES tables. This keeps the information of a region, it's associated calendar and the currency.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|---------------|----------|----------------|---|
| REGION_ID | NOT NULL | NUMBER | REGION_ID is the ID derived from PPM's KNTA_REGIONS table. |
| REGION_NAME | NOT NULL | VARCHAR2 (100) | Name of this region |
| DESCRIPTION | NULLABLE | VARCHAR2 (200) | Description for this region |
| CALENDAR_ID | NOT NULL | NUMBER | ID of the calendar associated with this region derived from PPM's KDRV_CALENDARS table. |
| CALENDAR_NAME | NULLABLE | VARCHAR2 (100) | Name of this calendar |
| CURRENCY_ID | NOT NULL | NUMBER | ID of the currency associated with this region, derived from PPM's KCST_CURRENCIES table. |
| CURRENCY_NAME | NOT NULL | VARCHAR2 (100) | Name of this currency |
| ENABLED_FLAG | NOT NULL | VARCHAR2 (1) | A flag to indicate if this region is enabled or disabled |

Indexes

This table uses no indexes.

Sequences

This table uses no sequences.

RPT_DIM_REQ_DTL_CUSTOM_PARAMS

This table holds detail custom parameters for request entity.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|------------------------|----------|-----------------|--|
| REQUEST_ID | NOT NULL | NUMBER | Request ID of the request |
| REQUEST_TYPE_ID | NOT NULL | NUMBER | The request type ID for this request |
| BATCH_NUMBER | NOT NULL | NUMBER | This column is used when the custom parameters are more than 50 and the 50 columns provided in the table is insufficient. This corresponds to the PPM table's batch number. For example, if the BATCH_NUMBER is 2 and the parameter number is 4, it is considered as parameter 54. |
| REQUEST_TYPE_NAME | NULLABLE | VARCHAR2 (80) | Name of this request type |
| PARAMETER1-40 | NULLABLE | VARCHAR2 (200) | Value of custom detail parameters used by the customer for this request |
| VISIBLE_PARAMETER1-40 | NULLABLE | VARCHAR2 (200) | Visible value of custom detail parameters used by the customer for this request |
| PARAMETER41-50 | NULLABLE | VARCHAR2 (4000) | Value of custom detail parameters used by the customer for this request |
| VISIBLE_PARAMETER41-50 | NULLABLE | VARCHAR2 (4000) | Visible value of custom detail parameters used by the customer for this request |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|----------------------------|------------|----------|-----------------|
| RPT_REQ_DTL_CSTM_PARAMS_N1 | NONUNIQUE | 1 | REQUEST_ID |
| RPT_REQ_DTL_CSTM_PARAMS_N2 | NONUNIQUE | 1 | REQUEST_TYPE_ID |

Sequences

This table uses no sequences.

RPT_DIM_REQ_HDR_CUSTOM_PARAMS

This table holds header custom parameters for request entities.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-------------------|----------|---------------|--|
| REQUEST_ID | NOT NULL | NUMBER | Request ID of the request |
| REQUEST_TYPE_ID | NOT NULL | NUMBER | The request type ID for this request |
| BATCH_NUMBER | NOT NULL | NUMBER | This column is used when the custom parameters are more than 50 and the 50 columns provided in the table is insufficient. This corresponds to the PPM table's batch number. For example, if the BATCH_NUMBER is 2 and the parameter number is 4, it is considered as parameter 54. |
| REQUEST_TYPE_NAME | NULLABLE | VARCHAR2 (80) | Name of this request type |

| Column Name | Null? | Data Type | Description |
|------------------------|----------|-----------------|---|
| PARAMETER1-40 | NULLABLE | VARCHAR2 (200) | Value of custom header parameters used by the customer for this request |
| VISIBLE_PARAMETER1-40 | NULLABLE | VARCHAR2 (200) | Visible value of custom header parameters used by the customer for this request |
| PARAMETER41-50 | NULLABLE | VARCHAR2 (4000) | Value of custom header parameters used by the customer for this request |
| VISIBLE_PARAMETER41-50 | NULLABLE | VARCHAR2 (4000) | Visible value of custom header parameters used by the customer for this request |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|----------------------------|------------|----------|-----------------|
| RPT_REQ_HDR_CSTM_PARAMS_N1 | NONUNIQUE | 1 | REQUEST_ID |
| RPT_REQ_HDR_CSTM_PARAMS_N2 | NONUNIQUE | 1 | REQUEST_TYPE_ID |

Sequences

This table uses no sequences.

RPT_DIM_REQUESTS

This table holds request information derived from PPM tables. All the fields are derived from PPM request tables.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|--------------------------------|----------|-----------------|---|
| SOURCE_ENTITY_TYPE | NULLABLE | VARCHAR2 (20) | The type of this request. Proposal, Project, Asset or Other |
| LIFECYCLE_ID | NULLABLE | NUMBER | This is the PFM entity's (proposal, project, or asset) lifecycle Id. This ID is referring to PPM's PFM_LIFECYCLE_PARENT_ENTITY table. |
| STAFFING_PROFILE_ID | NULLABLE | NUMBER | ID of staffing profile this entity is attached to |
| SOURCE_ENTITY_ID | NULLABLE | NUMBER | The request ID of this entity |
| SPAWNED_PROJECT_ID | NULLABLE | NUMBER | ID of the project that was spawned by this request. This ID refers to PPM's PM_PROJECTS table. |
| SOURCE_ENTITY_NAME | NULLABLE | VARCHAR2 (300) | Name of this entity. If proposal, it is proposal name, if project, it is project's name and so on... |
| SOURCE_ENTITY_HEALTH_MEANING | NULLABLE | VARCHAR2 (80) | Source entity's health meaning obtained from lookup table |
| SOURCE_ENTITY_HEALTH_CODE | NULLABLE | VARCHAR2 (30) | Source entity's health code |
| SOURCE_ENTITY_MANAGER_ID | NULLABLE | VARCHAR2 (4000) | User ID of the source entity's manager |
| SOURCE_BENEFIT_MANAGER_ID | NULLABLE | VARCHAR2 (4000) | User ID of the source entity's benefit manager. |
| SOURCE_ENTITY_FIN_SUMMARY_ID | NULLABLE | NUMBER | ID of financial summary that is attached to this entity |
| SOURCE_ENTITY_FIN_SUMMARY_NAME | NULLABLE | VARCHAR2 (1500) | Name of financial summary that is attached to this entity |

| Column Name | Null? | Data Type | Description |
|--|----------|-------------------|---|
| ENTITY_ PROJECT_ CLASS_ MEANING | NULLABLE | VARCHAR2 (80) | Project's class meaning derived from lookups |
| ENTITY_ PROJECT_ CLASS_ CODE | NULLABLE | VARCHAR2 (30) | Project's class code |
| ENTITY_ASSET_ CLASS_ MEANING | NULLABLE | VARCHAR2 (80) | Asset's class meaning derived from lookups |
| ENTITY_ASSET_ CLASS_ CODE | NULLABLE | VARCHAR2 (30) | Asset's class code |
| BUSINESS_ OBJECTIVE_ NAME | NULLABLE | VARCHAR2 (80) | Business objective name of this entity |
| BUSINESS_ OBJECTIVE_ ID | NULLABLE | NUMBER | Business objective ID of this entity |
| BUSINESS_ UNIT | NULLABLE | VARCHAR2 (80) | Business unit of this entity derived from lookups |
| BUSINESS_ UNIT_ CODE | NULLABLE | VARCHAR2 (30) | Business unit code of this entity |
| ENTITY_ REGION | NULLABLE | VARCHAR2 (100) | Name of the region this entity belongs to |
| ENTITY_ REGION_ ID | NULLABLE | NUMBER | ID of the region this entity belongs to |
| RETURN_ON_ INVESTMENT | NULLABLE | NUMBER | ROI of this request |
| NET_PRESENT_ VALUE | NULLABLE | NUMBER | NPV of this request |
| CUSTOM_ FIELD_ VALUE | NULLABLE | NUMBER | Financial metric for this entity |
| TOTAL_SCORE | NULLABLE | NUMBER | Total score of this entity |
| VALUE_RATING | NULLABLE | NUMBER | The score for score domain "Value" |
| RISK_RATING | NULLABLE | NUMBER | The score for score domain "Risk" |
| DISCOUNT_ RATE | NULLABLE | NUMBER | Discount rate for this entity |

| Column Name | Null? | Data Type | Description |
|---------------------------|----------|-----------------|---|
| REQUEST_ID | NOT NULL | NUMBER | Unique identifier for this request |
| REQUEST_TYPE_ID | NULLABLE | NUMBER | The type of this request |
| REQUEST_TYPE_NAME | NULLABLE | VARCHAR2 (80) | Name of this request type |
| ENTITY_STATUS | NULLABLE | VARCHAR2 (80) | Status of this entity |
| PRIORITY | NULLABLE | VARCHAR2 (150) | Priority of this entity derived from lookups table |
| DEPARTMENT | NULLABLE | VARCHAR2 (150) | Department of this entity derived from lookups table |
| APPLICATION | NULLABLE | VARCHAR2 (150) | Application of this entity derived from lookups table |
| WORKFLOW_NAME | NULLABLE | VARCHAR2 (150) | Name of the workflow associated with this entity |
| DESCRIPTION | NULLABLE | VARCHAR2 (3000) | Description given to this request |
| ACTIVE_WORKFLOW_STEP_NAME | NULLABLE | VARCHAR2 (2000) | Name of the active workflow step |
| CREATED_BY | NULLABLE | VARCHAR2 (200) | Identifier for the user who created the record |
| CREATION_DATE | NULLABLE | DATE | Date record was created |
| CONTACT_NAME | NULLABLE | VARCHAR2 (200) | The full name of the contact person for this request |
| CONTACT_PHONE | NULLABLE | VARCHAR2 (30) | The phone number for this contact person |
| CONTACT_EMAIL | NULLABLE | VARCHAR2 (80) | The email for this contact person |
| ASSIGNED_TO | NULLABLE | VARCHAR2 (200) | User ID that this request is assigned to |
| USER_DATA1-20 | NULLABLE | VARCHAR2 (200) | User data segment |

| Column Name | Null? | Data Type | Description |
|-----------------------|----------|----------------|--|
| VISIBLE_USER_DATA1-20 | NULLABLE | VARCHAR2 (200) | User data segment |
| DIRECT_PORTFOLIO_ID | NULLABLE | NUMBER | ID of the portfolio this request directly belongs to |
| DIRECT_PORTFOLIO_NAME | NULLABLE | VARCHAR2 (200) | Name of the portfolio this request directly belongs to |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|---------------------|------------|----------|--------------------|
| RPT_DIM_REQUESTS_N1 | NONUNIQUE | 1 | SOURCE_ENTITY_ID |
| RPT_DIM_REQUESTS_N2 | NONUNIQUE | 1 | SOURCE_ENTITY_TYPE |
| RPT_DIM_REQUEST_PK | UNIQUE | 1 | REQUEST_ID |

Sequences

This table uses no sequences.

RPT_DIM_RM_POSITIONS

This is a materialized view and stores the position information for a given staffing profile.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-----------------------|----------|----------------|--|
| POSITION_ID | NOT NULL | NUMBER | Unique identifier to represent this position and is derived from PPM RSC_POSTIONS table. |
| STAFFING_PROFILE_ID | NULLABLE | NUMBER | This is a foreign to the parent table RPT_DIM_RM_STAFF_PROFILES. Although there is no explicit foreign key constraint on this column. |
| RESOURCE_POOL_ID | NULLABLE | VARCHAR2 (40) | The resource pool ID from which the resources are requested to fill this position. |
| POSITION_NAME | NULLABLE | VARCHAR2 (255) | Name of this position. |
| ROLE_NAME | NOT NULL | VARCHAR2 (260) | The role selected for this position. Note the resource assigned to this position may have a different role than the role specified on this position. |
| USER_DATA1-20 | NULLABLE | VARCHAR2 (255) | User data segment |
| VISIBLE_USER_DATA1-20 | NULLABLE | VARCHAR2 (255) | User data segment |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|-------------------------|------------|----------|---------------|
| RPT_DIM_RM_POSITIONS_N4 | NONUNIQUE | 1 | POSITION_ID |
| RPT_DIM_RM_POSITIONS_N5 | NONUNIQUE | 1 | POSITION_NAME |

Sequences

This table uses no sequences.

RPT_DIM_RM_RESOURCEPOOLMGRS

This is an association table to establish the many-to-many relationship between the RPT_DIM_RM_RESOURCES and RPT_DIM_RM_RESOURCEPOOLS tables.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|---------------------------|----------|----------------|---|
| RESOURCE_POOL_ID | NOT NULL | NUMBER | System-generated identifier and points to the RPT_DIM_RM_RESOURCE_POOLS.RESOURCE_POOL_ID |
| RESOURCEPOOL_MGR_USR_ID | NOT NULL | NUMBER | System-generated identifier and points to the RPT_DIM_RM_RESOURCES.RESOURCE_ID |
| RESOURCE_POOL_NAME | NULLABLE | VARCHAR2 (260) | The resource pool name same as in RPT_DIM_RM_RESOURCE_POOLS.RESOURCE_POOL_NAME. The data in this column has been denormalized for reports/universe as many to many relationship is not supported in the universe. |
| RESOURCEPOOL_MANAGER_NAME | NULLABLE | VARCHAR2 (200) | The resource pool manager name (resource name) same as in RPT_DIM_RM_RESOURCES.RESOURCE_NAME. The data in this column has been denormalized for reports/universe as many to many relationship is not supported in the universe. |

Indexes

This table uses no indexes.

Sequences

This table uses no sequences.

RPT_DIM_RM_RESOURCEPOOLS

This is a materialized view on top of PPM's RSC_RESOURCE_POOLS table with additional logic to build the resource pool hierarchy.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-----------------------------|----------|-----------------|--|
| RANK | NULLABLE | NUMBER | Specifies what is the level of this resource pool in the hierarchy. The rank starts with number 1. 1 being the root. |
| RESOURCE_POOL_ID | NULLABLE | NUMBER | ID derived from the PPM RSC_RESOURCE_POOLS table. |
| RESOURCE_POOL_NAME | NULLABLE | VARCHAR2 (260) | Name of this resource pool derived from the underlying RSC_RESOURCE_POOLS table. |
| ORG_UNIT_ID | NULLABLE | NUMBER | ID of the org unit this resource pool is tied to. In the original PPM table, this represents a foreign key to the KRSC_ORG_UNIT table. |
| REGION_ID | NULLABLE | NUMBER | ID of the region this resource pool is tied to. In the original PPM table, this represents a foreign key to the KNTA_REGIONS table. |
| DIRECT_PARENT_RES_POOL_ID | NULLABLE | NUMBER | ID of the immediate parent resource pool |
| DIRECT_PARENT_RES_POOL_NAME | NULLABLE | VARCHAR2 (4000) | Name of the immediate parent resource pool |

| Column Name | Null? | Data Type | Description |
|----------------------------|----------|-----------------|---|
| PARENT_RESOURCE_POOL_ID1 | NULLABLE | VARCHAR2 (4000) | ID of the root resource pool in the hierarchy, level 1 or rank 1 |
| PARENT_RESOURCE_POOL_NAME1 | NULLABLE | VARCHAR2 (4000) | Name of the root resource pool in the hierarchy, level 1 or rank 1 |
| PARENT_RESOURCE_POOL_ID2 | NULLABLE | VARCHAR2 (4000) | ID of the root resource pool in the hierarchy, level 2 or rank 2 |
| PARENT_RESOURCE_POOL_NAME2 | NULLABLE | VARCHAR2 (4000) | Name of the root resource pool in the hierarchy, level 2 or rank 2 |
| PARENT_RESOURCE_POOL_ID3 | NULLABLE | VARCHAR2 (4000) | ID of the root resource pool in the hierarchy, level 3 or rank 3 |
| PARENT_RESOURCE_POOL_NAME3 | NULLABLE | VARCHAR2 (4000) | Name of the root resource pool in the hierarchy, level 3 or rank 3. |
| PARENT_RESOURCE_POOL_ID4 | NULLABLE | VARCHAR2 (4000) | ID of the root resource pool in the hierarchy, level 4 or rank 4. |
| PARENT_RESOURCE_POOL_NAME4 | NULLABLE | VARCHAR2 (4000) | Name of the root resource pool in the hierarchy, level 4 or rank 4. |
| PARENT_RESOURCE_POOL_ID5 | NULLABLE | VARCHAR2 (4000) | ID of the root resource pool in the hierarchy, level 5 or rank 5. |
| PARENT_RESOURCE_POOL_NAME5 | NULLABLE | VARCHAR2 (4000) | Name of the root resource pool in the hierarchy, level 5 or rank 5. |
| PARENT_RESOURCE_POOL_ID6 | NULLABLE | VARCHAR2 (4000) | ID of the root resource pool in the hierarchy, level 6 or rank 6. |
| PARENT_RESOURCE_POOL_NAME6 | NULLABLE | VARCHAR2 (4000) | Name of the root resource pool in the hierarchy, level 6 or rank 6. |

| Column Name | Null? | Data Type | Description |
|-----------------------------|----------|-----------------|---|
| PARENT_RESOURCE_POOL_ID7 | NULLABLE | VARCHAR2 (4000) | ID of the root resource pool in the hierarchy, level 7 or rank 7. |
| PARENT_RESOURCE_POOL_NAME7 | NULLABLE | VARCHAR2 (4000) | Name of the root resource pool in the hierarchy, level 7 or rank 7. |
| PARENT_RESOURCE_POOL_ID8 | NULLABLE | VARCHAR2 (4000) | ID of the root resource pool in the hierarchy, level 8 or rank 8. |
| PARENT_RESOURCE_POOL_NAME8 | NULLABLE | VARCHAR2 (4000) | Name of the root resource pool in the hierarchy, level 8 or rank 8. |
| PARENT_RESOURCE_POOL_ID9 | NULLABLE | VARCHAR2 (4000) | ID of the root resource pool in the hierarchy, level 9 or rank 9. |
| PARENT_RESOURCE_POOL_NAME9 | NULLABLE | VARCHAR2 (4000) | Name of the root resource pool in the hierarchy, level 9 or rank 9. |
| PARENT_RESOURCE_POOL_ID10 | NULLABLE | VARCHAR2 (4000) | ID of the root resource pool in the hierarchy, level 10 or rank 10. |
| PARENT_RESOURCE_POOL_NAME10 | NULLABLE | VARCHAR2 (4000) | Name of the root resource pool in the hierarchy, level 10 or rank 10. |
| USER_DATA1-20 | NULLABLE | VARCHAR2 (255) | User data segment |
| VISIBLE_USER_DATA1-20 | NULLABLE | VARCHAR2 (255) | User data segment |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|-----------------------------|------------|----------|--------------------|
| RPT_DIM_RM_RESOURCEPOOLS_N8 | NONUNIQUE | 1 | RESOURCE_POOL_ID |
| RPT_DIM_RM_RESOURCEPOOLS_N9 | NONUNIQUE | 1 | RESOURCE_POOL_NAME |

Sequences

This table uses no sequences.

RPT_DIM_RM_RESOURCES

This is a materialized view over PPM's RSC_RESOURCES, KNTA_USERS, KRSC_ORG_UNITS, RSC_ROLES, and so on. This is one of the dimensional tables in RM star schema.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-------------------|----------|----------------|---|
| RESOURCE_ID | NULLABLE | NUMBER | RESOURCE_ID derived from PPM's RSC_RESOURCES table |
| USER_NAME | NULLABLE | VARCHAR2 (200) | USER_NAME is the resource's username that is derived from KNTA_USERS table |
| FIRST_NAME | NULLABLE | VARCHAR2 (80) | FIRST_NAME is the resource's first name that is derived from KNTA_USERS table |
| LAST_NAME | NULLABLE | VARCHAR2 (80) | LAST_NAME is the resource's last name that is derived from KNTA_USERS table |
| EMAIL_ADDRESS | NULLABLE | VARCHAR2 (80) | EMAIL_ADDRESS is the resource's email address that is derived from KNTA_USERS table |
| PHONE_NUMBER | NULLABLE | VARCHAR2 (30) | PHONE_NUMBER is the resource's phone number that is derived from KNTA_USERS table |
| LOCATION_CODE | NULLABLE | VARCHAR2 (150) | LOCATION_CODE contains the meaning of resource's location from knta_lookups |
| RESOURCE_CATEGORY | NULLABLE | VARCHAR2 (150) | RESOURCE_CATEGORY contains the meaning of resource's category from knta_lookups |

| Column Name | Null? | Data Type | Description |
|---------------------------|----------|-----------------|---|
| DEPARTMENT | NULLABLE | VARCHAR2 (150) | DEPARTMENT contains the meaning of resource's department from knta_lookups |
| COMPANY | NULLABLE | VARCHAR2 (150) | COMPANY contains the meaning of resource's company from knta_lookups |
| TITLE | NULLABLE | VARCHAR2 (150) | TITLE contains the meaning of resource's title from knta_lookups |
| ROLE_NAME | NULLABLE | VARCHAR2 (260) | Resource's role name, denormalized. |
| PRIMARY_ROLE_ID | NULLABLE | NUMBER | Primary role ID of this resource. PPM just supports one role per resource in the application, although it says primary. |
| REGION_ID | NULLABLE | NUMBER | Region to which this resource is associated to |
| CALENDAR_NAME | NULLABLE | VARCHAR2 (100) | Name of the regional calendar (the calendar that is associated to resource's region) |
| START_DATE | NULLABLE | DATE | Start date for the resource. This is a required field for any resource in the system. |
| END_DATE | NULLABLE | DATE | End date of a resource and this is not a required field. |
| MANAGER_ID | NULLABLE | NUMBER | USER_ID of the resource's direct manager |
| MANAGER_FIRST_NAME | NULLABLE | VARCHAR2 (80) | Direct manager's first name |
| MANAGER_LAST_NAME | NULLABLE | VARCHAR2 (80) | Direct manager's last name |
| LEVEL1_MANAGER_ID | NULLABLE | VARCHAR2 (4000) | USER_ID of top level manager of this resource |
| LEVEL1_MANAGER_FIRST_NAME | NULLABLE | VARCHAR2 (4000) | FIRST_NAME of top level manager of this resource |
| LEVEL1_MANAGER_LAST_NAME | NULLABLE | VARCHAR2 (4000) | LAST_NAME of top level manager of this resource |
| LEVEL2_MANAGER_ID | NULLABLE | VARCHAR2 (4000) | USER_ID of level 2 manager of this resource |
| LEVEL2_MANAGER_FIRST_NAME | NULLABLE | VARCHAR2 (4000) | FIRST_NAME of level 2 manager of this resource |

| Column Name | Null? | Data Type | Description |
|-----------------------------------|----------|--------------------|--|
| LEVEL2_ MANAGER_ LAST_NAME | NULLABLE | VARCHAR2 (4000) | LAST_NAME of level 2 manager of this resource |
| LEVEL3_ MANAGER_ID | NULLABLE | VARCHAR2 (4000) | USER_ID of level 3 manager of this resource |
| LEVEL3_ MANAGER_ FIRST_NAME | NULLABLE | VARCHAR2 (4000) | FIRST_NAME of level 3 manager of this resource |
| LEVEL3_ MANAGER_ LAST_NAME | NULLABLE | VARCHAR2 (4000) | LAST_NAME of level 3 manager of this resource |
| LEVEL4_ MANAGER_ID | NULLABLE | VARCHAR2 (4000) | USER_ID of level 4 manager of this resource |
| LEVEL4_ MANAGER_ FIRST_NAME | NULLABLE | VARCHAR2 (4000) | FIRST_NAME of level 4 manager of this resource |
| LEVEL4_ MANAGER_ LAST_NAME | NULLABLE | VARCHAR2 (4000) | LAST_NAME of level 4 manager of this resource |
| LEVEL5_ MANAGER_ID | NULLABLE | VARCHAR2 (4000) | USER_ID of level 5 manager of this resource |
| LEVEL5_ MANAGER_ FIRST_NAME | NULLABLE | VARCHAR2 (4000) | FIRST_NAME of level 5 manager of this resource |
| LEVEL5_ MANAGER_ LAST_NAME | NULLABLE | VARCHAR2 (4000) | LAST_NAME of level 5 manager of this resource |
| LEVEL6_ MANAGER_ID | NULLABLE | VARCHAR2 (4000) | USER_ID of level 6 manager of this resource |
| LEVEL6_ MANAGER_ FIRST_NAME | NULLABLE | VARCHAR2 (4000) | FIRST_NAME of level 6 manager of this resource |
| LEVEL6_ MANAGER_ LAST_NAME | NULLABLE | VARCHAR2 (4000) | LAST_NAME of level 6 manager of this resource |
| LEVEL7_ MANAGER_ID | NULLABLE | VARCHAR2 (4000) | USER_ID of level 7 manager of this resource |

| Column Name | Null? | Data Type | Description |
|------------------------------------|----------|--------------------|--|
| LEVEL7_ MANAGER_ FIRST_NAME | NULLABLE | VARCHAR2 (4000) | FIRST_NAME of level 7 manager of this resource |
| LEVEL7_ MANAGER_ LAST_NAME | NULLABLE | VARCHAR2 (4000) | LAST_NAME of level 7 manager of this resource |
| LEVEL8_ MANAGER_ID | NULLABLE | VARCHAR2 (4000) | USER_ID of level 8 manager of this resource |
| LEVEL8_ MANAGER_ FIRST_NAME | NULLABLE | VARCHAR2 (4000) | FIRST_NAME of level 8 manager of this resource |
| LEVEL8_ MANAGER_ LAST_NAME | NULLABLE | VARCHAR2 (4000) | LAST_NAME of level 8 manager of this resource |
| LEVEL9_ MANAGER_ID | NULLABLE | VARCHAR2 (4000) | USER_ID of level 9 manager of this resource |
| LEVEL9_ MANAGER_ FIRST_NAME | NULLABLE | VARCHAR2 (4000) | FIRST_NAME of level 9 manager of this resource |
| LEVEL9_ MANAGER_ LAST_NAME | NULLABLE | VARCHAR2 (4000) | LAST_NAME of level 9 manager of this resource |
| LEVEL10_ MANAGER_ID | NULLABLE | VARCHAR2 (4000) | USER_ID of level 10 manager of this resource |
| LEVEL10_ MANAGER_ FIRST_NAME | NULLABLE | VARCHAR2 (4000) | FIRST_NAME of level 10 manager of this resource |
| LEVEL10_ MANAGER_ LAST_NAME | NULLABLE | VARCHAR2 (4000) | LAST_NAME of level 10 manager of this resource |
| PRIMARY_ORG_ UNIT_ID | NULLABLE | NUMBER | Resource's primary org unit |
| NAMED_ RESOURCE_ FLAG | NULLABLE | VARCHAR2 (1) | Flag to indicate if this resource is a named or unnamed resource |
| TM_APPROVER | NULLABLE | VARCHAR2 (4000) | Username of this resource's time approver |

| Column Name | Null? | Data Type | Description |
|-----------------------------|----------|-----------------|--|
| TM_APPROVER_SEC_GRP | NULLABLE | VARCHAR2 (4000) | The security group name who can approve the time sheet |
| TM_BILLING_APPROVER | NULLABLE | VARCHAR2 (4000) | Username of this resource's billing approver |
| TM_BILLING_APPROVER_SEC_GRP | NULLABLE | VARCHAR2 (4000) | The security group name for resource's billing approvals |
| USER_DATA1-20 | NULLABLE | VARCHAR2 (200) | User data segment |
| VISIBLE_USER_DATA1-20 | NULLABLE | VARCHAR2 (200) | User data segment |

Indexes

This table uses no indexes.

Sequences

This table uses no sequences.

RPT_DIM_RM_ROLES

This is a materialized view over PPM's RSC_ROLES table. It is a simple table with role's details.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-------------|----------|-----------|--|
| ROLE_ID | NOT NULL | NUMBER | ROLE_ID derived from PPM's RSC_ROLES table |

| Column Name | Null? | Data Type | Description |
|------------------|----------|-------------------|--|
| ROLE_NAME | NOT NULL | VARCHAR2 (260) | Name of this role |
| DESCRIPTION | NULLABLE | VARCHAR2 (650) | Description of this role |
| ENABLED_ FLAG | NULLABLE | VARCHAR2(1) | Flag to indicate if this role is enabled or disabled |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|----------------------|------------|----------|-------------|
| PKC_RSC_ROLES_NLS1 | UNIQUE | 1 | ROLE_ID |
| RPT_DIM_RM_ROLES_N14 | NONUNIQUE | 1 | ROLE_NAME |

Sequences

This table uses no sequences.

RPT_DIM_RM_STAFF_PROFILES

This is a materialized view that contains information from PPM's RSC_STAFFING_PROFILES table.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-------------------------|----------|-----------|--|
| STAFFING_ PROFILE_ID | NULLABLE | NUMBER | Unique identifier that determines a staffing profile and derived from PPM RSC_STAFFING_PROFILES table. |

| Column Name | Null? | Data Type | Description |
|-----------------------------|----------|-----------------|---|
| STAFFING_PROFILE_NAME | NULLABLE | VARCHAR2 (300) | Name of this staffing profile |
| REGION_ID | NULLABLE | NUMBER | Region attached to this staffing profile |
| DESCRIPTION | NULLABLE | VARCHAR2 (650) | Description of this staffing profile |
| ORG_UNIT_CONTAINER_ID | NULLABLE | NUMBER | Org_unit ID if the parent type of this staffing profile is org unit |
| PFM_ENTITY_CONTAINER_ID | NULLABLE | NUMBER | The PFM entity ID if the parent type of this staffing profile is PFM entity |
| CONTAINER_TYPE | NULLABLE | VARCHAR2 (150) | Staffing profile's parent type, like org unit or Proposal, Project or Asset |
| CONTAINER_NAME | NULLABLE | VARCHAR2 (4000) | Name of the staffing profile's parent or container |
| WORKLOAD_CATEGORY | NULLABLE | VARCHAR2 (4000) | Workload category of this staffing profile. This represents the meaning as in lookup table. |
| CREATED_ON | NULLABLE | DATE | Date when this staffing profile was created |
| CREATED_BY | NULLABLE | VARCHAR2 (200) | Identifier for the user who created the record |
| START_DATE | NULLABLE | DATE | Start_date of this staffing profile as mentioned in it's header |
| FINISH_DATE | NULLABLE | DATE | Finish date of this staffing profile as mentioned in it's header |
| IS_BASELINE_FLAG | NULLABLE | VARCHAR2 (150) | A flag to represent if this staffing profile is a base line staffing profile |
| IS_CONSIDERED_WORKLOAD_FLAG | NULLABLE | VARCHAR2 (150) | A flag to represent if this staffing profile is considered as workload category |
| STATUS_CODE | NULLABLE | NUMBER | Status code of this staffing profile |
| STATUS | NULLABLE | VARCHAR2 (150) | Status of this staffing profile. Meaning for the above status code for this SP |
| USER_DATA1-20 | NULLABLE | VARCHAR2 (255) | User data segment |
| VISIBLE_USER_DATA1-20 | NULLABLE | VARCHAR2 (255) | User data segment |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|------------------------------|------------|----------|-----------------------|
| RPT_DIM_RM_STAFF_PROFILES_N1 | NONUNIQUE | 1 | STAFFING_PROFILE_ID |
| RPT_DIM_RM_STAFF_PROFILES_N2 | NONUNIQUE | 1 | STAFFING_PROFILE_NAME |
| RPT_DIM_RM_STAFF_PROFILES_N3 | NONUNIQUE | 1 | IS_BASELINE_FLAG |

Sequences

This table uses no sequences.

RPT_DIM_TM_ACTIVITIES_MV

This table stores the definition of each PPM Activity.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|---------------|----------|----------------|--|
| ACTIVITY_ID | NOT NULL | NUMBER | ID derived from the PPM ITG_ACTIVITIES table |
| ACTIVITY_NAME | NOT NULL | VARCHAR2 (255) | Name of the activity |
| DESCRIPTION | NULLABLE | VARCHAR2 (650) | Description of the activity |
| ENABLED_FLAG | NULLABLE | VARCHAR2 (1) | Indicates whether the activity is enabled |

| Column Name | Null? | Data Type | Description |
|-------------------------|----------|----------------|--|
| CAPEX_OPEX_ENABLED_FLAG | NULLABLE | VARCHAR2 (1) | Indicates whether the Activity can be capitalized. Y indicates that the Activity can be capitalized. |
| CATEGORY | NULLABLE | VARCHAR2 (150) | Indicates the SOP 98-1 Category for the activity. |

Indexes

This table uses no indexes.

Sequences

This table uses no sequences.

RPT_DIM_TM_CHARGE_CODES_MV

This table stores the definition of each PPM charge code.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|----------------------|----------|----------------|--|
| CHARGE_CODE_ID | NOT NULL | NUMBER | ID derived from the PPM KTMG_CHARGE_CODES table. |
| CHARGE_CODE_NAME | NOT NULL | VARCHAR2 (80) | Name of the charge code |
| DESCRIPTION | NOT NULL | VARCHAR2 (240) | Description of the charge code |
| CHARGE_CODE_CATEGORY | NULLABLE | VARCHAR2 (150) | Category of the charge code |

| Column Name | Null? | Data Type | Description |
|--------------|----------|-------------------|---|
| DEPARTMENT | NULLABLE | VARCHAR2 (150) | Department of the charge code |
| CLIENT | NULLABLE | VARCHAR2 (150) | Client of the charge code |
| ENABLED_FLAG | NOT NULL | VARCHAR2 (1) | Indicates whether the charge code is enabled. |

Indexes

This table uses no indexes.

Sequences

This table uses no sequences.

RPT_DIM_TM_PERIODS_MV

This table defines the time periods used in Time Management.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|------------------|----------|------------------|---|
| PERIOD_ID | NOT NULL | NUMBER | ID derived from the PPM KTMG_PERIODS table. |
| PERIOD_NAME | NOT NULL | VARCHAR2 (80) | User defined name for the period. |
| PERIOD_TYPE_NAME | NOT NULL | VARCHAR2 (80) | Name of the period type |

| Column Name | Null? | Data Type | Description |
|-------------|----------|-----------|--|
| START_DATE | NOT NULL | DATE | Date on which the period starts (inclusive). |
| END_DATE | NOT NULL | DATE | Date on which the period ends (inclusive). |

Indexes

This table uses no indexes.

Sequences

This table uses no sequences.

RPT_DIM_TM_TIME_SHEET_LINES

This table stores the definition of each Time Sheet line within the ETL range derived from the PPM TM_TIME_SHEET_LINES table.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|--------------------|----------|----------------|--|
| TIME_SHEET_LINE_ID | NOT NULL | NUMBER | ID derived from the PPM TM_TIME_SHEET_LINES table |
| TIME_SHEET_ID | NOT NULL | VARCHAR2 (260) | Foreign key to the Time Sheet ID from RPT_DIM_TM_TIME_SHEETS table |
| STATE | NULLABLE | NUMBER | Time Sheet Line state. |

| Column Name | Null? | Data Type | Description |
|-------------------------|----------|----------------|--|
| WORK_ITEM_TYPE | NULLABLE | VARCHAR2 (255) | Indicates the type of item derived from the PPM TM_TIME_SHEET_LINES table. Valid values are REQUEST, PACKAGE, TASK, PROJECT, and MISC |
| WORK_ITEM_SET_TYPE | NULLABLE | VARCHAR2 (12) | Indicates the type of item set. Valid values are REQUEST_TYPE, WORKFLOW, PROJECT, and MISC |
| WORK_ITEM_ID | NULLABLE | VARCHAR2 (255) | Identifies the item that this line represents derived from the PPM TM_TIME_SHEET_LINES table. Valid values represent Request IDs, Package IDs, Task IDs, and Miscellaneous Items |
| WORK_ITEM_SET_ID | NULLABLE | VARCHAR2 (255) | Identifies the parent type of the item derived from the PPM TM_TIME_SHEET_LINES table. Valid values represent Request Types, Workflows, Projects, and Miscellaneous Items |
| TASK_ID | NULLABLE | VARCHAR2 (255) | Identifies the PPM Task against which time is logged for this line |
| REQUEST_ID | NULLABLE | VARCHAR2 (255) | Identifies the PPM Request against which time is logged for this line |
| PACKAGE_ID | NULLABLE | VARCHAR2 (255) | Identifies the PPM Package against which time is logged for this line |
| MISC_CODE | NULLABLE | VARCHAR2 (255) | Identifies the name of the PPM Miscellaneous Item against which time is logged for this line |
| REQUEST_TYPE_ID | NULLABLE | VARCHAR2 (255) | Identifies the PPM request type for the request work item, if it exists for the line |
| MASTER_PROJECT_ID | NULLABLE | VARCHAR2 (255) | Identifies the PPM Project for the Task or Project Work Item, if it exists for the line |
| ACTUAL_APPROVER_USER_ID | NULLABLE | VARCHAR2 (40) | Identifies the PPM Resource who approved this line. This is a foreign key into the RPT_DIM_RM_RESOURCES table |
| USER_DATA1-20 | NULLABLE | VARCHAR2 (255) | User data segment |
| VISIBLE_USER_DATA1-20 | NULLABLE | VARCHAR2 (255) | User data segment |

| Column Name | Null? | Data Type | Description |
|----------------------------|----------|----------------|--|
| WORK_ITEM_NAME | NULLABLE | VARCHAR2 (300) | Name of the work item entity against which time is logged for this line |
| WORK_ITEM_SET_NAME | NULLABLE | VARCHAR2 (300) | Name of the parent entity against which time is logged for this line |
| REQUEST_TYPE_NAME | NULLABLE | VARCHAR2 (80) | PPM Request Type against which time is logged for this line, the WORK_ITEM_TYPE is 'REQUEST' |
| TASK_NAME | NULLABLE | VARCHAR2 (300) | PPM Task Name against which time is logged for this line, the WORK_ITEM_TYPE is 'TASK' or 'PROJECT' |
| PROJECT_NAME | NULLABLE | VARCHAR2 (300) | PPM Project Name against which time is logged for this line, the WORK_ITEM_TYPE is 'TASK' or 'PROJECT' |
| PACKAGE_NUMBER | NULLABLE | VARCHAR2 (40) | PPM Package Number against which time is logged for this line, the WORK_ITEM_TYPE is 'PACKAGE' |
| CHARGE_CODE_ID | NULLABLE | NUMBER | Foreign key to RPT_DIM_TM_CHARGE_CODES_MV |
| CHARGE_PERCENT | NULLABLE | FLOAT | Percentage for Charge code |
| MISC_NAME | NULLABLE | VARCHAR2 (150) | PPM Miscellaneous Name against which time is logged for this line when WORK_ITEM_TYPE is 'MISC' |
| ACTUAL_APPROVER_FIRST_NAME | NULLABLE | VARCHAR2 (80) | First Name of the PPM Resource who approved the line |
| ACTUAL_APPROVER_LAST_NAME | NULLABLE | VARCHAR2 (80) | Last Name of the PPM Resource who approved the line |
| NLS_STATUS | NULLABLE | VARCHAR2 (150) | Status of the Time Sheet Line in the PPM definition language |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|-----------------------------|------------|----------|---------------|
| RPT_DIM_TM_TIME_SHEET_LIN_1 | NONUNIQUE | 1 | TIME_SHEET_ID |
| RPT_DIM_TM_TIME_SHEET_LIN_1 | NONUNIQUE | 2 | STATE |

Sequences

This table uses no sequences.

RPT_DIM_TM_TIME_SHEETS

This table stores the definition of each Time Sheet within the ETL range derived from the PPM TM_TIME_SHEETS table.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|---------------|----------|----------------|--|
| TIME_SHEET_ID | NOT NULL | NUMBER | ID derived from the PPM TM_TIME_SHEETS table |
| CREATION_DATE | NOT NULL | DATE | Date on which the time sheet was created. |
| CREATED_BY | NOT NULL | NUMBER | ID for the user derived who created the time sheet. |
| RESOURCE_ID | NOT NULL | NUMBER | ID for the user for whom the time sheet was created. This is a foreign key into the RPT_DIM_RM_RESOURCES |
| DESCRIPTION | NULLABLE | VARCHAR2 (650) | Description of the time sheet. |

| Column Name | Null? | Data Type | Description |
|-------------------|----------|----------------|--|
| PERIOD_ID | NULLABLE | NUMBER | ID for the period. This is a foreign key to the RPT_DIM_TM_PERIODS_MV |
| STATUS | NULLABLE | VARCHAR2 (150) | Status for the Time Sheet. |
| TIME_SHEET_NUMBER | NULLABLE | NUMBER | Time Sheet Number. |
| REQUIRED_FLAG | NULLABLE | VARCHAR2 (1) | Indicates whether a time sheet is required for this period based on the Time Sheet Policy for the resource |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|--------------------------|------------|----------|---------------|
| RPT_DIM_TM_TIME_SHEETS_1 | NONUNIQUE | 1 | TIME_SHEET_ID |
| RPT_DIM_TM_TIME_SHEETS_1 | NONUNIQUE | 2 | STATUS |

Sequences

This table uses no sequences.

RPT_DIM_WORKFLOW_STEPS

This is a materialized view on top of KWFL_WORKFLOW_STEPS_NLS and KWFL_WORKFLOWS_NLS tables, representing the relationship between enabled workflow steps and workflows that these steps belong to.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|---------------|----------|----------------|---|
| STEP_ID | NOT NULL | NUMBER | ID of the workflow step |
| STEP_NAME | NOT NULL | VARCHAR2 (80) | Name of the workflow step |
| WORKFLOW_NAME | NOT NULL | VARCHAR2 (150) | Name of the workflow the workflow step belongs to |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|----------------------------|------------|----------|-------------|
| RPT_DIM_WORKFLOW_STEPS_N11 | NONUNIQUE | 1 | STEP_ID |
| RPT_DIM_WORKFLOW_STEPS_N12 | NONUNIQUE | 1 | STEP_NAME |

Sequences

This table uses no sequences.

RPT_ETL_JOB

This table stores the detail information of the jobs ran for incremental ETL

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-----------------------|----------|----------------|--|
| ETL_JOB_ID | NOT NULL | NUMBER | Unique identifier to represent a scheduled job |
| ETL_NAME | NOT NULL | VARCHAR2 (64) | Name of the ETL identifying ETL process for different modules, for example, Common, Resource Management, Financial management or Time Management. |
| JOB_NAME | NOT NULL | VARCHAR2 (64) | Name of the ETL module |
| JOB_TYPE | NOT NULL | NUMBER | Numeric value representing job type auto(0) or manual(1) |
| BATCH_ID | NOT NULL | NUMBER | Unique ID that corresponds to an ETL execution which is comprised of processing ETL for multiple modules, for example, Common, Resource Management, Financial Management or Time Management. |
| ETL_WINDOW_START_TIME | NULLABLE | DATE | The timestamp that indicates the beginning mark of CDC subscription window used when processing changes during scheduled job execution for incremental ETL |
| ETL_WINDOW_END_TIME | NULLABLE | DATE | The timestamp that indicates the ending mark of CDC subscription window used when processing changes during scheduled job execution for incremental ETL |
| EXEC_START_TIME | NULLABLE | DATE | The timestamp when the specific ETL module started execution |
| EXEC_END_TIME | NULLABLE | DATE | The timestamp when the specific ETL module finished execution |
| STATUS | NOT NULL | VARCHAR2 (64) | Status of the individual ETL module indicating whether it completed, failed or running |
| DESCRIPTION | NULLABLE | VARCHAR2 (200) | Description of the steps executed successfully in ETL module |

Indexes

This table uses no indexes.

Sequences

This table uses no sequences.

RPT_EVENT_DEF

This table stores the various types of event that are logged by various steps during FULL ETL and incremental ETL process.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-------------------|----------|---------------|-------------------------------|
| EVENT_ID | NOT NULL | NUMBER | Identifier of the event type |
| EVENT_DESCRIPTION | NOT NULL | VARCHAR2(512) | Description of the event type |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|----------------|------------|----------|-------------|
| SYS_C003623227 | UNIQUE | 1 | EVENT_ID |

Sequences

This table uses no sequences.

RPT_EVENT_LOG

This table stores the events with error or information logged during various steps executed during the FULL ETL or incremental ETL.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|--------------|----------|-----------|---|
| EVENT_LOG_ID | NOT NULL | NUMBER | Identifier to represent an event log. It refers to the EVENT_LOG_ID in RPT_EVENT_LOG_DETAIL table |
| EVENT_TIME | NOT NULL | DATE | The timestamp when the event occurs |
| EVENT_ID | NOT NULL | NUMBER | Identifier to represent an event type. This refers to the EVENT_ID column in RPT_EVENT_DEF |
| ETL_JOB_ID | NULLABLE | NUMBER | Identifier to relate with ETL JOB. This refers to the ETL_JOB_ID column in RPT_ETL_JOB table |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|----------------|------------|----------|--------------|
| SYS_C003623231 | UNIQUE | 1 | EVENT_LOG_ID |

Sequences

This table uses no sequences.

RPT_EVENT_LOG_DETAIL

This table stores the detail logging information from various steps performed during FULL ETL and incremental ETL process. Information in this table can be used for debugging, troubleshooting problems that occur during FULL ETL or incremental ETL process.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|------------------|----------|--------------------|--|
| EVENT_ LOG_ID | NOT NULL | NUMBER | Unique identifier to represent an event log detail |
| EVENT_ TIME | NOT NULL | DATE | The time stamp indicating when the event occurred |
| MODULE_ NAME | NULLABLE | VARCHAR2 (512) | Name of the module that generated the event |
| FUNC_ NAME | NULLABLE | VARCHAR2 (256) | The SQL procedure name which logged the event |
| FILE_NAME | NULLABLE | VARCHAR2 (512) | Name of the file where the SQL procedure is defined that generated the event |
| LINE_NO | NULLABLE | NUMBER | Line number in the file where the SQL procedure has logged the event |
| MSG | NULLABLE | VARCHAR2 (1024) | The descriptive message of the error or information |

Indexes

This table uses no indexes.

Sequences

This table uses no sequences.

RPT_FCT_FM_APPROVED_BUDGETS

This is a fact table in reporting Star schema that contains the approved budget information of lifecycle entities, programs, org units, and portfolios. The approved budget information is stored at year

granularity.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|----------------------|----------|----------------|---|
| FACT_ID | NOT NULL | NUMBER | ID derived from FM_APPROVED_BUDGETS table |
| FINANCIAL_SUMMARY_ID | NOT NULL | NUMBER | ID of the financial summary derived from FM_FINANCIAL_SUMMARY table |
| DIRECT_PARENT_ID | NOT NULL | NUMBER | ID of proposal, project, asset, program, org unit with which this financial summary is associated. This ID is derived from PFM_LIFECYCLE_PARENT_ENTITY table for currently active entity. |
| DIRECT_PARENT_TYPE | NOT NULL | VARCHAR2 (20) | Type of parent entity (project, proposal, asset, program, org unit) |
| DIRECT_PARENT_NAME | NOT NULL | VARCHAR2 (500) | Name of a proposal, project, asset, program, or org unit associated with the financial summary |
| REGION_ID | NULLABLE | NUMBER | ID of the region with which the cost is associated with derived from |
| PERIOD_ID | NOT NULL | NUMBER | ID of the period with which the cost is associated with derived from FM_APPROVED_BUDGETS table |
| AMOUNT_BASE | NULLABLE | NUMBER | Approved cost aggregated for a given year, displayed in base currency |
| AMOUNT_LOCAL | NULLABLE | NUMBER | Approved cost aggregated for a given year, displayed in local currency |
| EXPENSE_TYPE_CODE | NULLABLE | VARCHAR2 (40) | Indicates the expense type (capital or operating) |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|----------------|------------|----------|-------------|
| SYS_C003624083 | UNIQUE | 1 | FACT_ID |

Sequences

This table uses no sequences.

RPT_FCT_FM_BENEFIT_CELLS

This table holds benefit information derived from PPM tables. All the fields are derived from PPM benefit tables.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|----------------------|----------|-----------|--|
| FACT_ID | NOT NULL | NUMBER | The financial line cell ID that is derived from KNTA_USERS table |
| BENEFIT_ID | NOT NULL | NUMBER | The benefit ID that is derived from FM_BENEFIT_LINES table |
| BENEFIT_LINE_ID | NOT NULL | NUMBER | The benefit line ID that is derived from FM_BENEFIT_LINES table |
| FINANCIAL_SUMMARY_ID | NOT NULL | NUMBER | Financial summary ID that is derived from FM_FINANCIAL_SUMMARY table |
| DIRECT_PARENT_ID | NOT NULL | NUMBER | ID of the item where current financial information comes from |

| | | | |
|--------------------|----------|----------------|---|
| DIRECT_PARENT_TYPE | NOT NULL | VARCHAR2 (20) | Type of the item where current financial information comes from (for example, PROJECT/ ASSET/ PROPOSAL/ PROGRAM/ PORTFOLIO) |
| DIRECT_PARENT_NAME | NOT NULL | VARCHAR2 (500) | Name of the item where current financial information comes from |
| REGION_ID | NULLABLE | NUMBER | ID of the Region |
| PERIOD_ID | NOT NULL | NUMBER | Period ID that is derived from FM_FINANCIAL_LINE_CELLS table |
| ACTUAL_BASE | NULLABLE | NUMBER | Actual value in base currency that is derived from FM_FINANCIAL_LINE_CELLS table |
| ACTUAL_LOCAL | NULLABLE | NUMBER | Actual value in local currency that is derived from FM_FINANCIAL_LINE_CELLS table |
| PLANNED_BASE | NULLABLE | NUMBER | Planned value in base currency that is derived from FM_FINANCIAL_LINE_CELLS table |
| PLANNED_LOCAL | NULLABLE | NUMBER | Planned value in local currency that is derived from FM_FINANCIAL_LINE_CELLS table |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|---------------------------|------------|----------|----------------------|
| RPT_FCT_FM_BENEFIT_CLL_N1 | NONUNIQUE | 1 | PERIOD_ID |
| RPT_FCT_FM_BENEFIT_CLL_N2 | NONUNIQUE | 1 | DIRECT_PARENT_ID |
| RPT_FCT_FM_BENEFIT_CLL_N3 | NONUNIQUE | 1 | BENEFIT_LINE_ID |
| RPT_FCT_FM_BENEFIT_CLL_N4 | NONUNIQUE | 1 | FINANCIAL_SUMMARY_ID |
| RPT_FCT_FM_BENEFIT_CLL_N5 | NONUNIQUE | 1 | REGION_ID |
| RPT_FCT_FM_BENEFIT_CLL_N6 | NONUNIQUE | 1 | DIRECT_PARENT_TYPE |

Sequences

This table uses no sequences.

RPT_FCT_FM_FA_APPROVED_MV

This is a fact table in reporting Star schema that contains the approved budget, forecast and actual information of financial summary of lifecycle entities, programs, portfolios, and org units. This information is stored at year granularity.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|----------------------|----------|----------------|--|
| FINANCIAL_SUMMARY_ID | NULLABLE | NUMBER | ID of a financial summary derived from RPT_FCT_FM_APPROVED_BUDGETS table. |
| FORECAST_ACTUAL_ID | NULLABLE | NUMBER | Forecast Actual ID for a financial summary. |
| DIRECT_PARENT_ID | NULLABLE | NUMBER | ID of the financial entity that this financial summary belongs to (Project / Proposal / Asset / Program). |
| DIRECT_PARENT_TYPE | NULLABLE | VARCHAR2 (20) | Proposal, Project, Asset, Program |
| DIRECT_PARENT_NAME | NULLABLE | VARCHAR2 (500) | Name of the financial entity (project, proposal, asset, or program) to which this financial summary belongs. |
| REGION_ID | NULLABLE | NUMBER | ID of the region with which the cost is associated |
| PERIOD_ID | NULLABLE | NUMBER | ID of the Proposal/Project/Asset/Program/org units to which this Financial Summary belongs |
| APPROVED_BASE | NULLABLE | NUMBER | Approved cost aggregated for a given year, displayed in base currency |
| APPROVED_LOCAL | NULLABLE | NUMBER | Approved cost aggregated for a given year, displayed in local currency |
| FORECAST_BASE | NULLABLE | NUMBER | Forecast cost aggregated for a given year, displayed in base currency |

| Column Name | Null? | Data Type | Description |
|----------------------|----------|-----------|--|
| FORECAST_LOCAL | NULLABLE | NUMBER | Forecast cost aggregated for a given year, displayed in local currency |
| ACTUAL_BASE | NULLABLE | NUMBER | Actual cost aggregated for a given year, displayed in base currency |
| ACTUAL_LOCAL | NULLABLE | NUMBER | Actual cost aggregated for a given year, displayed in local currency |
| CAPEX_APPROVED_BASE | NULLABLE | NUMBER | Approved capital cost aggregated for a given year, displayed in base currency |
| CAPEX_APPROVED_LOCAL | NULLABLE | NUMBER | Approved capital cost aggregated for a given year, displayed in local currency |
| OPEX_APPROVED_BASE | NULLABLE | NUMBER | Approved operating cost aggregated for a given year, displayed in local currency |
| OPEX_APPROVED_LOCAL | NULLABLE | NUMBER | Approved operating cost aggregated for a given year, displayed in base currency |

Indexes

This table uses no indexes.

Sequences

This table uses no sequences.

RPT_FCT_FM_FCST_ACTUAL_CELL

This is a fact table in reporting Star schema that contains the forecast and actual information of financial summary of lifecycle entities, programs, and portfolios. This information is stored at month granularity.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-------------------------|----------|----------------|---|
| FACT_ID | NOT NULL | NUMBER | ID derived from the FM_FINANCIAL_LINE_CELLS table |
| FORECAST_ACTUAL_LINE_ID | NOT NULL | NUMBER | ID of each forecast actual line in a financial summary derived from FM_FORECAST_ACTUAL_LINES table |
| FORECAST_ACTUAL_ID | NOT NULL | NUMBER | Forecast Actual ID for a specific financial summary derived from FM_FORECAST_ACTUAL_LINES table |
| FINANCIAL_SUMMARY_ID | NOT NULL | NUMBER | ID for a financial summary the cost are associated with derived from FM_FINANCIAL_SUMMARY |
| DIRECT_PARENT_ID | NOT NULL | NUMBER | ID for the proposal, project, asset, program, org unit with which this financial summary is associated with. This ID is derived from PFM_LIFECYCLE_PARENT_ENTITY table for current active entity. |
| DIRECT_PARENT_TYPE | NOT NULL | VARCHAR2 (20) | Type of parent entity (project, proposal, asset, program, org unit) |
| DIRECT_PARENT_NAME | NOT NULL | VARCHAR2 (500) | Name of a proposal, project, asset, program, or org unit associated with the financial summary |
| REGION_ID | NULLABLE | NUMBER | ID of the region with which the cost is associated |
| PERIOD_ID | NOT NULL | NUMBER | ID of the region with which the cost is associated derived from FM_FINANCIAL_LINE_CELLS table |
| ACTUAL_BASE | NULLABLE | NUMBER | Actual cost aggregated for a given year, displayed in base currency |
| ACTUAL_LOCAL | NULLABLE | NUMBER | Actual cost aggregated for a given year, displayed in base currency |
| PLANNED_BASE | NULLABLE | NUMBER | Forecast cost aggregated for a given year, displayed in base currency |
| PLANNED_LOCAL | NULLABLE | NUMBER | Forecast cost aggregated for a given year, displayed in local currency |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|-----------------------------|------------|----------|-------------------------|
| RPT_FCT_FM_FCST_ACTL_CLL_N1 | NONUNIQUE | 1 | PERIOD_ID |
| RPT_FCT_FM_FCST_ACTL_CLL_N2 | NONUNIQUE | 1 | DIRECT_PARENT_ID |
| RPT_FCT_FM_FCST_ACTL_CLL_N3 | NONUNIQUE | 1 | FORECAST_ACTUAL_LINE_ID |
| RPT_FCT_FM_FCST_ACTL_CLL_N4 | NONUNIQUE | 1 | FINANCIAL_SUMMARY_ID |
| RPT_FCT_FM_FCST_ACTL_CLL_N5 | NONUNIQUE | 1 | REGION_ID |
| RPT_FCT_FM_FCST_ACTL_CLL_N6 | NONUNIQUE | 1 | DIRECT_PARENT_TYPE |

Sequences

This table uses no sequences.

RPT_FCT_PM_AGGR_PROJ_EFFORT

This table stores all the efforts for this project at the project level.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-----------------------------|----------|-----------|---|
| EFFORT_ID | NOT NULL | NUMBER | System-generated identifier by the sequence RPT_FCT_PM_AGGR_PROJ_EFFORT_S |
| PROJECT_ID | NULLABLE | NUMBER | Unique identifier that identifies this project |
| WORK_PLAN_ID | NULLABLE | NUMBER | The work plan identifier of this project |
| ACTUAL_EFFORT | NULLABLE | FLOAT | Actual effort aggregated at the project level |
| ASSIGNED_SCHEDULED_EFFORT | NULLABLE | FLOAT | Assigned scheduled effort aggregated at the project level |
| UNASSIGNED_SCHEDULED_EFFORT | NULLABLE | FLOAT | Unassigned scheduled effort aggregated at the project level |
| ESTIMATED_REMAINING_EFFORT | NULLABLE | FLOAT | The estimated remaining effort aggregated at the project level |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|-----------------------------|------------|----------|--------------|
| RPT_FCT_PM_AGGR_PRJ_EFRT_N1 | NONUNIQUE | 1 | PROJECT_ID |
| RPT_FCT_PM_AGGR_PRJ_EFRT_N2 | NONUNIQUE | 1 | WORK_PLAN_ID |

Sequences

This table uses no sequences.

RPT_FCT_PM_AGGR_TASK_EFFORT

This table stores the effort information in a workplan aggregated by task hierarchy. Effort information at a summary task level is an aggregated value of all tasks below it in the hierarchy and its own value (if there is any). At leaf task level effort information is non aggregated.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-----------------------------|----------|-----------|--|
| EFFORT_ID | NOT NULL | NUMBER | System-generated identifier by the sequence RPT_FCT_PM_AGGR_TASK_EFFORT_S |
| TASK_ID | NULLABLE | NUMBER | Unique identifier of the task in a project |
| PROJECT_ID | NULLABLE | NUMBER | ID that identifies the project to which this task belongs to |
| WORK_PLAN_ID | NULLABLE | NUMBER | ID that identifies the workplan to which this task belongs to |
| ACTUAL_EFFORT | NULLABLE | FLOAT | Actual effort aggregated at the task level according to task hierarchy. For leaf task the effort information is non aggregated value |
| ASSIGNED_SCHEDULED_EFFORT | NULLABLE | FLOAT | Assigned scheduled effort aggregated at the task level according to task hierarchy. For leaf task the effort information is non aggregated value. |
| UNASSIGNED_SCHEDULED_EFFORT | NULLABLE | FLOAT | Unassigned scheduled effort aggregated at the task level according to task hierarchy. For leaf task the effort information is non aggregated value. |
| ESTIMATED_REMAINING_EFFORT | NULLABLE | FLOAT | The estimated remaining effort aggregated at the task level according to task hierarchy. For leaf task the effort information is non aggregated value. |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|------------------------------|------------|----------|--------------|
| RPT_FCT_PM_AGGR_TASK_EFRT_N1 | NONUNIQUE | 1 | PROJECT_ID |
| RPT_FCT_PM_AGGR_TASK_EFRT_N2 | NONUNIQUE | 1 | WORK_PLAN_ID |
| RPT_FCT_PM_AGGR_TASK_EFRT_N3 | NONUNIQUE | 1 | TASK_ID |

Sequences

This table uses no sequences.

RPT_FCT_PM_TASK_ASSIGNMENTS

This table stores the actual effort, assigned schedule effort and estimated remaining effort information of tasks by individual resource level.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-----------------|----------|-----------|--|
| ASSIGNMENT_ID | NOT NULL | NUMBER | System-generated identifier by the sequence RPT_FCT_PM_TASK_ASSIGNMENT_S |
| PROJECT_ID | NOT NULL | NUMBER | ID of the project to which all these assignment records belong to |
| WORK_PLAN_ID | NOT NULL | NUMBER | ID of the workplan to which all these assignment records belong to |
| TASK_ID | NOT NULL | NUMBER | ID of the task to which all these assignment records belong to |
| ACT_REGION_ID | NULLABLE | NUMBER | ID of the region the actual effort is associated with. This is derived from the resource's region ID if the actual effort is logged by an assigned resource or contributor. If the actual effort is logged with null resource ID, the region is derived from Project's region ID |
| SCHED_REGION_ID | NULLABLE | NUMBER | Project's region ID to which the scheduled effort belongs to |

| Column Name | Null? | Data Type | Description |
|-----------------|----------|-----------|---|
| ROLE_ID | NULLABLE | NUMBER | ID of the role associated with task (if any), otherwise the primary role of the assigned resource. This can be null when neither task has a role assigned to it or the assigned resource does not have a primary role configured. |
| RESOURCE_ID | NULLABLE | NUMBER | The resource against whom the efforts are assigned/logged. Resource ID can be null. |
| ACT_START_DATE | NULLABLE | DATE | Actual start date of the task by this resource |
| ACT_FINISH_DATE | NULLABLE | DATE | Actual finish date of the task by this resource |
| ACT_DURATION | NULLABLE | FLOAT | Actual duration is number of days to complete this task |
| ACT_EFFORT | NULLABLE | FLOAT | Actual number of hours for this task by this resource |
| SCHED_EFFORT | NULLABLE | FLOAT | The scheduled effort in hours for this task and this resource. Scheduled effort is null when the resource ID is null |
| EST_FINISH_DATE | NULLABLE | DATE | The estimated finish date of this task for this resource |
| EST_REM_EFFORT | NULLABLE | FLOAT | The estimated remaining effort of this task for this resource |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|--------------------------------|------------|----------|-----------------|
| RPT_FCT_PM_TASK_ASSIGNMENTS_B1 | NONUNIQUE | 1 | PROJECT_ID |
| RPT_FCT_PM_TASK_ASSIGNMENTS_B2 | NONUNIQUE | 1 | WORK_PLAN_ID |
| RPT_FCT_PM_TASK_ASSIGNMENTS_B3 | NONUNIQUE | 1 | ACT_REGION_ID |
| RPT_FCT_PM_TASK_ASSIGNMENTS_B4 | NONUNIQUE | 1 | SCHED_REGION_ID |
| RPT_FCT_PM_TASK_ASSIGNMENTS_B5 | NONUNIQUE | 1 | ROLE_ID |
| RPT_FCT_PM_TASK_ASSIGNMENTS_B6 | NONUNIQUE | 1 | RESOURCE_ID |
| RPT_FCT_PM_TASK_ASSIGNMENTS_N1 | NONUNIQUE | 1 | ASSIGNMENT_ID |
| RPT_FCT_PM_TASK_ASSIGNMENTS_N2 | NONUNIQUE | 1 | TASK_ID |

Sequences

This table uses no sequences.

RPT_FCT_PM_TASK_COST

This is a fact table which stores the project cost information.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|--------------------|----------|-----------|--|
| TASK_COST_ID | NOT NULL | NUMBER | System-generated identifier using RPT_FCT_PM_TASK_COST_S sequence |
| TASK_ID | NOT NULL | NUMBER | ID of a task of a project |
| WORK_PLAN_ID | NULLABLE | NUMBER | ID of work plan of a project |
| PROJECT_ID | NULLABLE | NUMBER | ID of a PPM project. It relates to the project_id of the pm_projects table. |
| REGION_ID | NULLABLE | NUMBER | ID of the region the cost is associated with. It is derived from project's region. |
| PLAN_CAP_LABOR_BSE | NULLABLE | NUMBER | Planned capital labor cost of a project, expressed in base currency. |
| PLAN_CAP_LABOR_LCL | NULLABLE | NUMBER | Planned capital labor cost of a project, expressed in local currency. |

| Column Name | Null? | Data Type | Description |
|--|--------------|------------------|---|
| PLAN_ CAP_ NON_ LABOR_ BSE | NULLABLE | NUMBER | Planned capital non labor cost of a project, expressed in base currency. |
| PLAN_ CAP_ NON_ LABOR_ LCL | NULLABLE | NUMBER | Planned capital non labor cost of a project, expressed in local currency. |
| PLAN_OP_ LABOR_ BSE | NULLABLE | NUMBER | Planned operating labor cost of a project, expressed in base currency. |
| PLAN_OP_ LABOR_ LCL | NULLABLE | NUMBER | Planned operating labor cost of a project, expressed in local currency. |
| PLAN_OP_ NON_ LABOR_ BSE | NULLABLE | NUMBER | Planned operating non labor cost of a project, expressed in base currency. |
| PLAN_OP_ NON_ LABOR_ LCL | NULLABLE | NUMBER | Planned operating non labor cost of a project, expressed in local currency. |
| ACT_CAP_ LABOR_ BSE | NULLABLE | NUMBER | Actual capital labor cost of a project, expressed in base currency. |
| ACT_CAP_ LABOR_ LCL | NULLABLE | NUMBER | Actual capital labor cost of a project, expressed in local currency. |
| ACT_CAP_ NON_ LABOR_ BSE | NULLABLE | NUMBER | Actual capital non labor cost of a project, expressed in base currency. |
| ACT_CAP_ NON_ LABOR_ LCL | NULLABLE | NUMBER | Actual capital non labor cost of a project, expressed in local currency. |

| Column Name | Null? | Data Type | Description |
|----------------------|--------------|------------------|--|
| ACT_OP_LABOR_BSE | NULLABLE | NUMBER | Actual operating labor cost of a project, expressed in base currency. |
| ACT_OP_LABOR_LCL | NULLABLE | NUMBER | Actual operating labor cost of a project, expressed in local currency. |
| ACT_OP_NON_LABOR_BSE | NULLABLE | NUMBER | Actual operating non labor cost of a project, expressed in base currency. |
| ACT_OP_NON_LABOR_LCL | NULLABLE | NUMBER | Actual operating non labor cost of a project, expressed in local currency. |
| EARNED_VALUE_BSE | NULLABLE | NUMBER | Earned value of the completed project work, expressed in base currency. |
| EARNED_VALUE_LCL | NULLABLE | NUMBER | Earned value of the completed project work, expressed in local currency. |
| PLANNED_VALUE_BSE | NULLABLE | NUMBER | Planned value of work to be performed for a project, expressed in base currency. |
| PLANNED_VALUE_LCL | NULLABLE | NUMBER | Planned value of work to be performed for a project, expressed in local currency. |
| SPI | NULLABLE | FLOAT | Schedule Performance Index (ratio of earned value to planned value for a project), is a measure of project efficiency. An SPI equal to or greater than one indicates a favorable condition and value of less than one indicates an unfavorable condition. |
| CPI | NULLABLE | FLOAT | Schedule Performance Index (ratio of earned value to actual cost for a project), is a measure of cost efficiency of a project. A CPI equal to or greater than one indicates a favorable condition and value of less than one indicates an unfavorable condition. |

| Column Name | Null? | Data Type | Description |
|------------------|----------|----------------|--|
| BASE_CUR_CODE | NULLABLE | VARCHAR2 (255) | Base currency code associated with the project. Example: USD for United States Dollar. (The base currency defaults to the PPM Center base currency.) |
| BASE_CUR_SYMBOL | NULLABLE | VARCHAR2 (255) | Symbol for the base currency used for the project. Example: \$ for USD. |
| LOCAL_CUR_CODE | NULLABLE | VARCHAR2 (255) | Base currency code associated with the project. Example: USD for United States Dollar. (The local currency defaults to the PPM Center local currency.) |
| LOCAL_CUR_SYMBOL | NULLABLE | VARCHAR2 (255) | Symbol for the local currency used for the project. Example: \$ for USD. |

Indexes

This table uses no indexes.

Sequences

| Sequence Name | Sequence Type |
|------------------------|---------------|
| RPT_FCT_PM_TASK_COST_S | TASK_COST_ID |

RPT_FCT_PM_TASK_UNASSGN Effort

This table stores the unassigned scheduled effort at the task level without aggregation. There is a column RESOURCE_ID, which always contains NULL value and acts as a dummy column to help align this fact table with all other effort information facts which are available at the resource granularity.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-----------------------------|----------|-----------|---|
| EFFORT_ID | NOT NULL | NUMBER | System-generated identifier by the sequence RPT_FCT_PM_TASK_UNASG_EFRT_S |
| PROJECT_ID | NULLABLE | NUMBER | ID of a PPM project |
| WORK_PLAN_ID | NULLABLE | NUMBER | ID of a work plan of a project |
| TASK_ID | NULLABLE | NUMBER | ID of a task of a work plan |
| REGION_ID | NULLABLE | NUMBER | ID of the region unassigned scheduled effort is associated with. This is derived from project's region ID. |
| ROLE_ID | NULLABLE | NUMBER | ID of the task's role (if configured for the task), otherwise NULL |
| RESOURCE_ID | NULLABLE | NUMBER | This is a dummy column kept for aligning this fact table with all other effort fact information table at resource granularity. It always contains NULL value. |
| UNASSIGNED_SCHEDULED_EFFORT | NULLABLE | FLOAT | Unassigned effort in hours |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|--------------------------------|------------|----------|--------------|
| RPT_FCT_PM_TASK_UNASSG_EFRT_N1 | NONUNIQUE | 1 | TASK_ID |
| RPT_FCT_PM_TASK_UNASSG_EFRT_N2 | NONUNIQUE | 1 | PROJECT_ID |
| RPT_FCT_PM_TASK_UNASSG_EFRT_N3 | NONUNIQUE | 1 | WORK_PLAN_ID |
| RPT_FCT_PM_TASK_UNASSG_EFRT_N4 | NONUNIQUE | 1 | REGION_ID |
| RPT_FCT_PM_TASK_UNASSG_EFRT_N5 | NONUNIQUE | 1 | ROLE_ID |

Sequences

This table uses no sequences.

RPT_FCT_REQUEST_TRANSACTIONS

This table holds workflow transaction history derived from PPM tables. All the fields are derived from PPM workflow transaction tables.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|---------------------|----------|---------------|---|
| STEP_TRANSACTION_ID | NOT NULL | NUMBER | ID of the step transaction to which this history row belongs |
| REQUEST_ID | NULLABLE | NUMBER | ID of the request |
| TIME_INTERVAL | NULLABLE | NUMBER | Time interval between the first step and the final step |
| FIRST_STEP_DATE | NULLABLE | DATE | Date of the first step transaction |
| FINAL_STEP_DATE | NULLABLE | DATE | Date of the last step transaction |
| USER_ID | NULLABLE | NUMBER | User ID derived from KWFL_STEP_TRANSACTION_HISTORY.LAST_UPDATED_BY |
| STEP_IDENTIFIER | NULLABLE | VARCHAR2(200) | ID of the step transaction ID format: Sort Order-Step ID. For example, 2-1001-3-5004-6-7003 |
| WORKFLOW_STEP_ID | NULLABLE | NUMBER | ID of the workflow step that needs to be processed |
| FIRST_STEP_STATUS | NULLABLE | VARCHAR2(30) | Status of the first step transaction |
| FINAL_STEP_STATUS | NULLABLE | VARCHAR2(30) | Status of the final step transaction |

| Column Name | Null? | Data Type | Description |
|---------------------|----------|---------------|--|
| RESULT_VALUE | NULLABLE | VARCHAR2(200) | Result of the step transaction when this history row was created |
| VISIBLE_ERROR_VALUE | NULLABLE | VARCHAR2(200) | Error value of the step transaction when this history row was created |
| ERROR_MESSAGE | NULLABLE | VARCHAR2(240) | Error message for the step transaction when this history row was created |
| USER_COMMENT | NULLABLE | VARCHAR2(240) | Comments within the step transaction when this history row was created |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|--------------------------------|------------|----------|---------------------|
| FCT_REQUEST_TRANSACTIONS_IDX_1 | NONUNIQUE | 1 | REQUEST_ID |
| FCT_REQUEST_TRANSACTIONS_IDX_2 | NONUNIQUE | 1 | USER_ID |
| FCT_REQUEST_TRANSACTIONS_IDX_3 | NONUNIQUE | 1 | WORKFLOW_STEP_ID |
| SYS_C009015322 | UNIQUE | 1 | STEP_TRANSACTION_ID |

Sequences

This table uses no sequences.

RPT_FCT_RM_RES_DISTRIBUTION

This table stores the current percentage distribution of resources in the resource pools.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|----------------------|----------|-----------|---|
| RESOURCE_ID | NOT NULL | NUMBER | ID derived from the PPM RSC_RESOURCE_POOLS table |
| RESOURCE_POOL_ID | NOT NULL | NUMBER | ID derived from the PPM RSC_RESOURCE_POOLS table |
| DISTRIBUTION_DATE | NOT NULL | DATE | Date on which this resource pool distribution was last determined |
| DISTRIBUTION_PERCENT | NOT NULL | NUMBER | The percentage that effort is distributed to this resource in the resource pool |

Indexes

This table uses no indexes.

Sequences

This table uses no sequences.

RPT_FCT_RM_RESOURCE_CAPACITY

This is a fact table in reporting Star schema that contains the capacity information of resources in resource pools. The capacity information is stored at day granularity.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|---------------------|----------|----------------|---|
| CAPACITY_ID | NOT NULL | NUMBER | System-generated identifier using RPT_FCT_RM_RESOURCE_CAPACITY_S sequence |
| RESOURCE_POOL_ID | NULLABLE | NUMBER | ID of the resource pool the capacity information belongs to, derived from PPM RSC_RESOURCE_POOLS table. This ID refers to the RESOURCE_POOL_ID column in RPT_DIM_RM_RESOURCEPOOLS materialized view. |
| CAPACITY | NULLABLE | NUMBER | Capacity of a resource expressed in hours |
| CAPACITY_PERS_DAYS | NULLABLE | NUMBER | Capacity of a resource expressed in person days |
| CALENDAR_ID | NULLABLE | NUMBER | ID of the resource's regional calendar for capacity of named resources and ID of the resource pool's regional calendar for capacity of unnamed resources. This ID is derived from PPM KDRV_CALENDARS table. |
| ROLE_ID | NULLABLE | NUMBER | ID of the resource's primary role derived from PPM RSC_ROLES_NLS table. This ID refers to the ROLE_ID column in RPT_DIM_RM_ROLES materialized view |
| REGION_ID | NULLABLE | NUMBER | ID of the region the resource pool belongs to derived from PPM KNTA_REGIONS_NLS table. This ID refers to the REGION_ID column of RPT_DIM_REGIONS_MV materialized view |
| ASSIGNMENT_DATE | NULLABLE | DATE | Date against which capacity of a resource is stored |
| RESOURCE_ID | NULLABLE | VARCHAR2 (260) | ID of the resource derived from RSC_RESOURCES table. This ID refers to the RESOURCE_ID column in RPT_DIM_RM_RESOURCES materialized view. |
| STAFFING_PROFILE_ID | NULLABLE | NUMBER | ID of the Staffing profile. This column always has the value -1. It is used for querying resource capacity and resource demand fact together against the staffing profile dimension. |

| Column Name | Null? | Data Type | Description |
|--------------|----------|----------------|---|
| DATA_ORIGIN | NULLABLE | VARCHAR2 (256) | This column indicates whether the capacity information was populated in this table during FULL ETL process or incremental ETL process |
| ETL_DATETIME | NULLABLE | DATE | The timestamp when the row was inserted or updated |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|-------------------------------|------------|----------|------------------|
| RPT_FCT_RM_RSRC_CAP_ID_IDX | NONUNIQUE | 1 | CAPACITY_ID |
| RPT_FCT_RM_RSRC_CPCTY_X01 | NONUNIQUE | 1 | RESOURCE_POOL_ID |
| RPT_FCT_RM_RSRC_CPCTY_X03 | NONUNIQUE | 1 | REGION_ID |
| RPT_FCT_RM_RSRC_CPCTY_X04 | NONUNIQUE | 1 | RESOURCE_ID |
| RPT_FCT_RM_RSRC_CPCTY_X05 | NONUNIQUE | 1 | CALENDAR_ID |
| RPT_FCT_RM_RSRC_CPCTY_X06 | NONUNIQUE | 1 | ROLE_ID |
| RPT_FCT_RSC_CPCTY_ASS_DATE_X1 | NONUNIQUE | 1 | ASSIGNMENT_DATE |

Sequences

| Sequence Name | Sequence Type |
|--------------------------------|---------------|
| RPT_FCT_RM_RESOURCE_CAPACITY_S | CAPACITY_ID |

RPT_FCT_RM_RESOURCE_DEMAND

This is a fact table in reporting Star schema that contains the demand information of positions in staffing profile. The demand information is stored at day granularity.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|------------------------|----------|----------------|--|
| DEMAND_ID | NOT NULL | NUMBER | System-generated identifier using RPT_FCT_RM_RESOURCE_DEMAND_S |
| POSITION_ID | NULLABLE | NUMBER | ID of the position, derived from the PPM RSC_POSITIONS table. This ID refers to the RPT_DIM_RM_POSITIONS materialized view. |
| RESOURCE_ASSIGNMENT_ID | NULLABLE | NUMBER | ID of resource assignment derived from PPM RSC_RESOURCE_ASSIGNMENT table |
| RESOURCE_ID | NULLABLE | NUMBER | ID of resource derived from PPM RSC_RESOURCES table. This ID refers to RPT_DIM_RM_RESOURCES materialized view |
| RESOURCE_POOL_ID | NULLABLE | NUMBER | ID of resource pool derived from PPM RSC_RESOURCE_POOLS table. This ID refers to RPT_DIM_RM_RESOURCEPOOLS materialized view |
| STAFFING_PROFILE_ID | NULLABLE | NUMBER | ID of staffing profile derived from PPM RSC_STAFFING_PROFILES table. This ID refers to RPT_DIM_RM_STAFF_PROFILES materialized view |
| ASSIGNMENT_DATE | NULLABLE | DATE | Date against which demand is stored |
| DEMAND_STATUS | NULLABLE | VARCHAR2 (200) | Status of demand, can be 'Committed', 'Softbooked' or 'Unmet' |
| DEMAND | NULLABLE | NUMBER | Demand of a position expressed in hours |
| DEMAND_PERS_DAYS | NULLABLE | NUMBER | Demand of a position expressed in person days |
| ROLE_ID | NULLABLE | NUMBER | ID of role derived from PPM RSC_ROLES_NLS table. This ID refers to RPT_DIM_RM_ROLES materialized view. |
| REGION_ID | NULLABLE | NUMBER | ID of region derived from PPM KNTA_REGIONS_NLS table. This ID refers to RPT_DIM_RM_REGIONS_MV materialized view. |

| Column Name | Null? | Data Type | Description |
|-------------|----------|-----------|---|
| CALENDAR_ID | NULLABLE | NUMBER | ID of calendar derived from PPM KDRV_CALENDARS table. This represents the resource pool's regional calendar for Softbooked and Committed demand or staffing profile's regional calendar for Unmet demand. |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|-----------------------------|------------|----------|------------------------|
| RPT_FCT_RM_DMND_ASS_DATE_X1 | NONUNIQUE | 1 | ASSIGNMENT_DATE |
| RPT_FCT_RM_RSRC_DMND_ID_IDX | NONUNIQUE | 1 | DEMAND_ID |
| RPT_FCT_RM_RSRC_DMND_X01 | NONUNIQUE | 1 | RESOURCE_POOL_ID |
| RPT_FCT_RM_RSRC_DMND_X02 | NONUNIQUE | 1 | ROLE_ID |
| RPT_FCT_RM_RSRC_DMND_X03 | NONUNIQUE | 1 | REGION_ID |
| RPT_FCT_RM_RSRC_DMND_X04 | NONUNIQUE | 1 | RESOURCE_ID |
| RPT_FCT_RM_RSRC_DMND_X05 | NONUNIQUE | 1 | STAFFING_PROFILE_ID |
| RPT_FCT_RM_RSRC_DMND_X06 | NONUNIQUE | 1 | RESOURCE_ASSIGNMENT_ID |
| RPT_FCT_RM_RSRC_DMND_X07 | NONUNIQUE | 1 | DEMAND_STATUS |
| RPT_FCT_RM_RSRC_DMND_X08 | NONUNIQUE | 1 | POSITION_ID |
| RPT_FCT_RM_RSRC_DMND_X09 | NONUNIQUE | 1 | CALENDAR_ID |

Sequences

| Sequence Name | Sequence Type |
|------------------------------|---------------|
| RPT_FCT_RM_RESOURCE_DEMAND_S | DEMAND_ID |

RPT_FCT_RM_RESOURCE_DEMAND_STG

This is a staging table used for holding the changed demand fact information during incremental ETL process and used to merge the changes to the demand fact table RPT_FCT_RM_RESOURCE_DEMAND. This table has the same structure as the RPT_FCT_RM_RESOURCE_DEMAND table.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|------------------------|----------|----------------|---|
| DEMAND_ID | NOT NULL | NUMBER | System-generated identifier using RPT_FCT_RM_RESOURCE_DEMAND_S |
| POSITION_ID | NULLABLE | NUMBER | ID of the position, derived from from the PPM RSC_POSITIONS table. This ID refers to the RPT_DIM_RM_POSITIONS materialized view. |
| RESOURCE_ASSIGNMENT_ID | NULLABLE | NUMBER | ID of resource assignment derived from PPM RSC_RESOURCE_ASSIGNMENT table |
| RESOURCE_ID | NULLABLE | NUMBER | ID of resource derived from PPM RSC_RESOURCES table. This ID refers to RPT_DIM_RM_RESOURCES materialized view. |
| RESOURCE_POOL_ID | NULLABLE | NUMBER | ID of resource pool derived from PPM RSC_RESOURCE_POOLS table. This ID refers to RPT_DIM_RM_RESOURCEPOOLS materialized view. |
| STAFFING_PROFILE_ID | NULLABLE | NUMBER | ID of staffing profile derived from PPM RSC_STAFFING_PROFILES table. This ID refers to RPT_DIM_RM_STAFF_PROFILES materialized view. |
| ASSIGNMENT_DATE | NULLABLE | DATE | Date against which demand is stored |
| DEMAND_STATUS | NULLABLE | VARCHAR2 (200) | Status of demand, can be 'Committed', 'Softbooked' or 'Unmet' |

| Column Name | Null? | Data Type | Description |
|------------------|----------|-----------|---|
| DEMAND | NULLABLE | NUMBER | Demand of a position expressed in hours |
| DEMAND_PERS_DAYS | NULLABLE | NUMBER | Demand of a position expressed in person days |
| ROLE_ID | NULLABLE | NUMBER | ID of role derived from PPM RSC_ROLES_NLS table. This ID refers to RPT_DIM_RM_ROLES materialized view. |
| REGION_ID | NULLABLE | NUMBER | ID of region derived from PPM KNTA_REGIONS_NLS table. This ID refers to RPT_DIM_RM_REGIONS_MV materialized view. |
| CALENDAR_ID | NULLABLE | NUMBER | ID of calendar derived from PPM KDRV_CALENDARS table. This represents the resource pool's regional calendar for Softbooked and Committed demand or staffing profile's regional calendar for Unmet demand. |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|-----------------------------|------------|----------|------------------------|
| RPT_FCT_RM_DMND_ASS_DATE_X2 | NONUNIQUE | 1 | ASSIGNMENT_DATE |
| RPT_STG_RM_RSRC_DMND_X01 | NONUNIQUE | 1 | RESOURCE_POOL_ID |
| RPT_STG_RM_RSRC_DMND_X02 | NONUNIQUE | 1 | ROLE_ID |
| RPT_STG_RM_RSRC_DMND_X03 | NONUNIQUE | 1 | REGION_ID |
| RPT_STG_RM_RSRC_DMND_X04 | NONUNIQUE | 1 | RESOURCE_ID |
| RPT_STG_RM_RSRC_DMND_X05 | NONUNIQUE | 1 | STAFFING_PROFILE_ID |
| RPT_STG_RM_RSRC_DMND_X06 | NONUNIQUE | 1 | RESOURCE_ASSIGNMENT_ID |
| RPT_STG_RM_RSRC_DMND_X07 | NONUNIQUE | 1 | DEMAND_STATUS |
| RPT_STG_RM_RSRC_DMND_X08 | NONUNIQUE | 1 | POSITION_ID |
| RPT_STG_RM_RSRC_DMND_X09 | NONUNIQUE | 1 | CALENDAR_ID |

Sequences

This table uses no sequences.

RPT_FCT_RM_RESOURCE_EFFORT

This table stores the actual effort information of a resource in hours, person days and FTE unit pre-aggregated by different period granularity, for example, week, month, quarter, half year or year.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|---------------------------|----------|-----------|--|
| RESOURCE_EFFORT_ID | NOT NULL | NUMBER | System-generated identifier using RPT_FCT_RM_RESOURCE_EFFORT_S |
| PERIOD_TYPE | NULLABLE | NUMBER | Indicates whether the period is a week, month, quarter, half year or year. |
| PERIOD_START_DATE | NULLABLE | DATE | Start date of a period derived from PPM PPM_FISCAL_PERIODS_NLS table. |
| PERIOD_END_DATE | NULLABLE | DATE | End date of a period derived from PPM PPM_FISCAL_PERIODS_NLS table. |
| PERIOD_EFFORT_HRS | NULLABLE | NUMBER | Aggregated actual effort information for a period expressed in hours. |
| PERIOD_EFFORT_PERSON_DAYS | NULLABLE | NUMBER | Aggregated actual effort information for a period expressed in person days. |
| PERIOD_EFFORT_FTE | NULLABLE | NUMBER | Aggregated actual effort information for a period expressed in FTE. |
| PROJECT_ID | NULLABLE | NUMBER | ID of the project against which actual effort is logged. It is derived from PPM PM_PROJECTS table and refers to RPT_DIM_PROJECTS_MV materialized view. |

| Column Name | Null? | Data Type | Description |
|---------------------|----------|-----------|---|
| REQUEST_ID | NULLABLE | NUMBER | ID of the request against which actual effort is logged. It is derived from PPM KCRT_REQUESTS table and refers to RPT_DIM_REQUESTS dimension table. |
| RESOURCE_ID | NULLABLE | NUMBER | ID of resource derived from PPM RSC_RESOURCES table. This ID refers to RPT_DIM_RM_RESOURCES materialized view. |
| ROLE_ID | NULLABLE | NUMBER | ID of role derived from PPM RSC_ROLES_NLS table. This ID refers to RPT_DIM_RM_ROLES materialized view. This represents task's role, if the task against which time logged has role associated otherwise it refers to the resource's primary role. |
| RESOURCE_POOL_ID | NULLABLE | NUMBER | ID of resource pool derived from PPM RSC_RESOURCE_POOLS table. This ID refers to RPT_DIM_RM_RESOURCEPOOLS materialized view. |
| STAFFING_PROFILE_ID | NULLABLE | NUMBER | ID of staffing profile derived from PPM RSC_STAFFING_PROFILES table. This ID refers to RPT_DIM_RM_STAFF_PROFILES materialized view. |
| REGION_ID | NULLABLE | NUMBER | ID of region derived from PPM KNTA_REGIONS_NLS table. This ID refers to RPT_DIM_RM_REGIONS_MV materialized view. This represents region the resource logging time belongs to. |
| CALENDAR_ID | NULLABLE | NUMBER | ID of calendar derived from PPM KDRV_CALENDARS table. This represents the resource's regional calendar. |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|-------------------------------|------------|----------|--------------------|
| RPT_IDX_RM_FCT_EFFRT_CAL_ID | NONUNIQUE | 1 | CALENDAR_ID |
| RPT_IDX_RM_FCT_EFFRT_END_DATE | NONUNIQUE | 1 | PERIOD_END_DATE |
| RPT_IDX_RM_FCT_EFFRT_ID | NONUNIQUE | 1 | RESOURCE_EFFORT_ID |
| RPT_IDX_RM_FCT_EFFRT_PROJ_ID | NONUNIQUE | 1 | PROJECT_ID |
| RPT_IDX_RM_FCT_EFFRT_REGN_ID | NONUNIQUE | 1 | REGION_ID |

| Index Name | Index Type | Sequence | Column Name |
|--------------------------------|------------|----------|---------------------|
| RPT_IDX_RM_FCT_EFFRT_REQ_ID | NONUNIQUE | 1 | REQUEST_ID |
| RPT_IDX_RM_FCT_EFFRT_RESRC_ID | NONUNIQUE | 1 | RESOURCE_ID |
| RPT_IDX_RM_FCT_EFFRT_ROLE_ID | NONUNIQUE | 1 | ROLE_ID |
| RPT_IDX_RM_FCT_EFFRT_RP_ID | NONUNIQUE | 1 | RESOURCE_POOL_ID |
| RPT_IDX_RM_FCT_EFFRT_SP_ID | NONUNIQUE | 1 | STAFFING_PROFILE_ID |
| RPT_IDX_RM_FCT_EFFRT_STRT_DATE | NONUNIQUE | 1 | PERIOD_START_DATE |

Sequences

| Sequence Name | Sequence Type |
|------------------------------|--------------------|
| RPT_FCT_RM_RESOURCE_EFFORT_S | RESOURCE_EFFORT_ID |

RPT_FCT_RM_RESRCE_CAPACITY_STG

This is a staging table used for holding the changed capacity fact information during incremental ETL process and used to merge the changes to the capacity fact table RPT_FCT_RM_RESOURCE_CAPACITY. This table has the same structure as the RPT_FCT_RM_RESOURCE_CAPACITY table.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-------------|----------|-----------|---|
| CAPACITY_ID | NOT NULL | NUMBER | System-generated identifier using RPT_FCT_RM_RESOURCE_CAPACITY_S sequence |

| Column Name | Null? | Data Type | Description |
|---------------------|----------|----------------|---|
| RESOURCE_POOL_ID | NULLABLE | NUMBER | ID of the resource pool the capacity information belongs to, derived from PPM RSC_RESOURCE_POOLS table. This ID refers to the RESOURCE_POOL_ID column in RPT_DIM_RM_RESOURCEPOOLS materialized view. |
| CAPACITY | NULLABLE | NUMBER | Capacity of a resource expressed in hours. |
| CAPACITY_PERS_DAYS | NULLABLE | NUMBER | Capacity of a resource expressed in person days. |
| CALENDAR_ID | NULLABLE | NUMBER | ID of the resource's regional calendar for capacity of named resources and ID of the resource pool's regional calendar for capacity of unnamed resources. This ID is derived from PPM KDRV_CALENDARS table. |
| ROLE_ID | NULLABLE | NUMBER | ID of the resource's primary role derived from PPM RSC_ROLES_NLS table. This ID refers to the ROLE_ID column in RPT_DIM_RM_ROLES materialized view |
| REGION_ID | NULLABLE | NUMBER | ID of the region the resource pool belongs to derived from PPM KNTA_REGIONS_NLS table. This ID refers to the REGION_ID column of RPT_DIM_REGIONS_MV materialized view |
| ASSIGNMENT_DATE | NULLABLE | DATE | Date against which capacity of a resource is stored. |
| RESOURCE_ID | NULLABLE | VARCHAR2 (260) | ID of the resource derived from RSC_RESOURCES table. This ID refers to the RESOURCE_ID column in RPT_DIM_RM_RESOURCES materialized view. |
| STAFFING_PROFILE_ID | NULLABLE | NUMBER | ID of the Staffing profile. This column always has the value -1. It is used for querying resource capacity and resource demand fact together against the staffing profile dimension. |
| DATA_ORIGIN | NULLABLE | VARCHAR2 (256) | This column indicates whether the capacity information was populated in this table during FULL ETL process or incremental ETL process. |
| ETL_DATETIME | NULLABLE | DATE | The timestamp when the row was inserted or updated |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|-------------------------------|------------|----------|------------------|
| RPT_FCT_RSC_CPCTY_ASS_DATE_X2 | NONUNIQUE | 1 | ASSIGNMENT_DATE |
| RPT_STG_RM_RSRC_CPCTY_X01 | NONUNIQUE | 1 | RESOURCE_POOL_ID |
| RPT_STG_RM_RSRC_CPCTY_X03 | NONUNIQUE | 1 | REGION_ID |
| RPT_STG_RM_RSRC_CPCTY_X04 | NONUNIQUE | 1 | RESOURCE_ID |
| RPT_STG_RM_RSRC_CPCTY_X05 | NONUNIQUE | 1 | CALENDAR_ID |
| RPT_STG_RM_RSRC_CPCTY_X06 | NONUNIQUE | 1 | ROLE_ID |

Sequences

This table uses no sequences.

RPT_FCT_RM_RSC_CAPACITY_AGGR

This is an aggregate table which stores the capacity information in hours, person days and FTE unit pre-aggregated by different period granularity, for example, week, month, quarter, half year and year.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|--------------|----------|-----------|---|
| CAPACITY_ID | NOT NULL | NUMBER | System-generated identifier using RPT_FCT_RM_RSC_CAPACITY_AGGR_S sequence |
| NUM_WRK_DAYS | NULLABLE | NUMBER | Number of working days in a period for example, week, month, quarter, half year or year |

| Column Name | Null? | Data Type | Description |
|---------------------|----------|-----------|---|
| PERIOD_HOURS | NULLABLE | NUMBER | Aggregated capacity information for a period expressed in hours. |
| PERIOD_PERS_DAYS | NULLABLE | NUMBER | Aggregated capacity information for a period expressed in person days. |
| PERIOD_FTE | NULLABLE | NUMBER | Aggregated capacity information for a period expressed in FTE. |
| PERIOD_TYPE | NULLABLE | NUMBER | Indicates whether the period is a week, month, quarter, half year or year. |
| START_DATE | NULLABLE | DATE | Start date of a period derived from PPM PPM_FISCAL_PERIODS_NLS table. |
| END_DATE | NULLABLE | DATE | End date of a period derived from PPM PPM_FISCAL_PERIODS_NLS table |
| CALENDAR_ID | NULLABLE | NUMBER | ID of the resource's regional calendar for capacity of named resources and ID of the resource pool's regional calendar for capacity of unnamed resources. This ID is derived from PPM KDRV_CALENDARS table. |
| RESOURCE_ID | NULLABLE | NUMBER | ID of the resource derived from RSC_RESOURCES table. This ID refers to the RESOURCE_ID column in RPT_DIM_RM_RESOURCES materialized view. |
| RESOURCE_POOL_ID | NULLABLE | NUMBER | ID of the resource pool the capacity information belongs to, derived from PPM RSC_RESOURCE_POOLS table. This ID refers to the RESOURCE_POOL_ID column in RPT_DIM_RM_RESOURCEPOOLS materialized view. |
| ROLE_ID | NULLABLE | NUMBER | ID of the resource's primary role derived from PPM RSC_ROLES_NLS table. This ID refers to the ROLE_ID column in RPT_DIM_RM_ROLES materialized view. |
| REGION_ID | NULLABLE | NUMBER | ID of the region the resource pool belongs to derived from PPM KNTA_REGIONS_NLS table. This ID refers to the REGION_ID column of RPT_DIM_REGIONS_MV materialized view. |
| STAFFING_PROFILE_ID | NULLABLE | NUMBER | ID of the Staffing profile. This column always has the value -1. It is used for querying resource capacity and resource demand fact together against the staffing profile dimension. |
| ETL_DATETIME | NULLABLE | DATE | The timestamp when the row was inserted or updated |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|--------------------------------|------------|----------|------------------|
| RPT_FCT_RM_RSC_CAP_AGGR_U1 | UNIQUE | 1 | CAPACITY_ID |
| RPT_FCT_RM_RSRC_CPCTY_AGGR_X01 | NONUNIQUE | 1 | RESOURCE_ID |
| RPT_FCT_RM_RSRC_CPCTY_AGGR_X02 | NONUNIQUE | 1 | RESOURCE_POOL_ID |
| RPT_FCT_RM_RSRC_CPCTY_AGGR_X03 | NONUNIQUE | 1 | ROLE_ID |
| RPT_FCT_RM_RSRC_CPCTY_AGGR_X04 | NONUNIQUE | 1 | REGION_ID |
| RPT_FCT_RM_RSRC_CPCTY_AGGR_X05 | NONUNIQUE | 1 | CALENDAR_ID |
| RPT_FCT_RM_RSRC_CPCTY_AGGR_X06 | NONUNIQUE | 1 | START_DATE |

Sequences

| Sequence Name | Sequence Type |
|--------------------------------|---------------|
| RPT_FCT_RM_RSC_CAPACITY_AGGR_S | CAPACITY_ID |

RPT_FCT_RM_RSC_DEMAND_AGGR

This is an aggregate table which stores the demand information in hours, person days and FTE unit pre-aggregated by different period granularity, for example, week, month, quarter, half year and year.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|------------------|----------|----------------|--|
| DEMAND_ID | NOT NULL | NUMBER | System-generated identifier using RPT_FCT_RM_RSC_DEMAND_AGGR_S |
| NUM_WRK_DAYS | NULLABLE | NUMBER | Number of working days in a period for example, week, month, quarter, half year or year |
| PERIOD_FTE | NULLABLE | NUMBER | Aggregated demand information for a period expressed in FTE. |
| PERIOD_PERS_DAYS | NULLABLE | NUMBER | Aggregated demand information for a period expressed in person days. |
| PERIOD_HOURS | NULLABLE | NUMBER | Aggregated demand information for a period expressed in hours. |
| PERIOD_TYPE | NULLABLE | NUMBER | Indicates whether the period is a week, month, quarter, half year or year. |
| START_DATE | NULLABLE | DATE | Start date of a period derived from PPM PPM_FISCAL_PERIODS_NLS table. |
| END_DATE | NULLABLE | DATE | End date of a period derived from PPM PPM_FISCAL_PERIODS_NLS table |
| DEMAND_STATUS | NULLABLE | VARCHAR2 (200) | Status of demand, can be 'Committed', 'Softbooked' or 'Unmet' |
| ASSIGNMENT_ID | NULLABLE | NUMBER | ID of resource assignment derived from PPM RSC_RESOURCE_ASSIGNMENT table |
| POSITION_ID | NULLABLE | NUMBER | ID of the position, derived from from the PPM RSC_POSITIONS table. This ID refers to the RPT_DIM_RM_POSITIONS materialized view. |
| RESOURCE_ID | NULLABLE | NUMBER | ID of resource derived from PPM RSC_RESOURCES table. This ID refers to RPT_DIM_RM_RESOURCES materialized view. |
| RESOURCE_POOL_ID | NULLABLE | NUMBER | ID of resource pool derived from PPM RSC_RESOURCE_POOLS table. This ID refers to RPT_DIM_RM_RESOURCEPOOLS materialized view. |

| Column Name | Null? | Data Type | Description |
|---------------------|----------|-----------|---|
| STAFFING_PROFILE_ID | NULLABLE | NUMBER | ID of staffing profile derived from PPM RSC_STAFFING_PROFILES table. This ID refers to RPT_DIM_RM_STAFF_PROFILES materialized view. |
| ROLE_ID | NULLABLE | NUMBER | ID of role derived from PPM RSC_ROLES_NLS table. This ID refers to RPT_DIM_RM_ROLES materialized view. |
| REGION_ID | NULLABLE | NUMBER | ID of region derived from PPM KNTA_REGIONS_NLS table. This ID refers to RPT_DIM_RM_REGIONS_MV materialized view. |
| CALENDAR_ID | NULLABLE | NUMBER | ID of calendar derived from PPM KDRV_CALENDARS table. This represents the resource pool's regional calendar for Softbooked and Committed demand or staffing profile's regional calendar for Unmet demand. |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|-------------------------------|------------|----------|---------------------|
| RPT_FCT_RM_RSC_DEM_AGGG_U1 | UNIQUE | 1 | DEMAND_ID |
| RPT_FCT_RM_RSRC_DMND_AGGG_X01 | NONUNIQUE | 1 | RESOURCE_POOL_ID |
| RPT_FCT_RM_RSRC_DMND_AGGG_X02 | NONUNIQUE | 1 | ROLE_ID |
| RPT_FCT_RM_RSRC_DMND_AGGG_X03 | NONUNIQUE | 1 | REGION_ID |
| RPT_FCT_RM_RSRC_DMND_AGGG_X04 | NONUNIQUE | 1 | RESOURCE_ID |
| RPT_FCT_RM_RSRC_DMND_AGGG_X05 | NONUNIQUE | 1 | STAFFING_PROFILE_ID |
| RPT_FCT_RM_RSRC_DMND_AGGG_X06 | NONUNIQUE | 1 | ASSIGNMENT_ID |
| RPT_FCT_RM_RSRC_DMND_AGGG_X07 | NONUNIQUE | 1 | DEMAND_STATUS |

| Index Name | Index Type | Sequence | Column Name |
|-------------------------------|------------|----------|-------------|
| RPT_FCT_RM_RSRC_DMND_AGGR_X08 | NONUNIQUE | 1 | POSITION_ID |
| RPT_FCT_RM_RSRC_DMND_AGGR_X09 | NONUNIQUE | 1 | CALENDAR_ID |
| RPT_FCT_RM_RSRC_DMND_AGGR_X10 | NONUNIQUE | 1 | START_DATE |

Sequences

| Sequence Name | Sequence Type |
|------------------------------|---------------|
| RPT_FCT_RM_RSC_DEMAND_AGGR_S | DEMAND_ID |

RPT_FCT_TM_ACTUAL_EFFORT

This table stores work item actual effort information.

Foreign Keys

This table does not have any foreign keys.

Column Description

| Column Name | Null? | Data Type | Description |
|-----------------|----------|-----------|--|
| EFFORT_ID | NOT NULL | NUMBER | System-generated identifier by the sequence RPT_FCT_TM_ACTUAL_EFFORT_S |
| EFFORT_DATE | NOT NULL | DATE | Date for which the effort was submitted |
| EFFORT_HRS | NULLABLE | NUMBER | The number of effort hours entered for this date |
| TASK_ID | NULLABLE | NUMBER | ID of the Task if the effort is logged against a Task |
| TASK_PROJECT_ID | NULLABLE | NUMBER | ID of the Project if the Work Item is a Task or a Project |
| PROJECT_ID | NULLABLE | NUMBER | ID of the Project if the effort is logged against a Project |

| Column Name | Null? | Data Type | Description |
|-------------------------|----------|----------------|---|
| REQUEST_ID | NULLABLE | NUMBER | ID of the request if the effort is logged against a request |
| PACKAGE_ID | NULLABLE | NUMBER | ID of the Package if the effort is logged against a Package |
| MISCELLANEOUS_ITEM_NAME | NULLABLE | VARCHAR2 (150) | The Name of the Miscellaneous Item if the effort is logged against a Miscellaneous Item |
| RESOURCE_ID | NULLABLE | NUMBER | ID of the Resource who logged the effort |
| TIME_SHEET_LINE_ID | NULLABLE | NUMBER | ID of the Time Sheet Line where the effort was logged |
| ACTIVITY_ID | NULLABLE | NUMBER | ID of the Activity associated with this Time Sheet Line |
| TASK_ROLE_ID | NULLABLE | NUMBER | The Role assigned to a Task if the effort is logged against a Task |
| RESOURCE_POOL_ID | NULLABLE | NUMBER | ID of the resource pool to which the resource is assigned |
| DISTRIBUTION_PERCENT | NULLABLE | NUMBER | The percentage of effort distributed to this resource pool |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|------------------------------|------------|----------|----------------------|
| RPT_FCT_TM_ACTUAL_EFFORT_B1 | NONUNIQUE | 1 | DISTRIBUTION_PERCENT |
| RPT_FCT_TM_ACTUAL_EFFORT_B10 | NONUNIQUE | 1 | RESOURCE_POOL_ID |
| RPT_FCT_TM_ACTUAL_EFFORT_B11 | NONUNIQUE | 1 | REQUEST_ID |
| RPT_FCT_TM_ACTUAL_EFFORT_B12 | NONUNIQUE | 1 | TASK_PROJECT_ID |
| RPT_FCT_TM_ACTUAL_EFFORT_B7 | NONUNIQUE | 1 | RESOURCE_ID |
| RPT_FCT_TM_ACTUAL_EFFORT_B8 | NONUNIQUE | 1 | ACTIVITY_ID |

| Index Name | Index Type | Sequence | Column Name |
|-----------------------------|------------|----------|--------------------|
| RPT_FCT_TM_ACTUAL_EFFORT_B9 | NONUNIQUE | 1 | TASK_ROLE_ID |
| RPT_FCT_TM_ACTUAL_EFFORT_N1 | NONUNIQUE | 1 | TIME_SHEET_LINE_ID |
| SYS_C003624095 | UNIQUE | 1 | EFFORT_ID |

Sequences

| Sequence Name | Sequence Type |
|----------------------------|---------------|
| RPT_FCT_TM_ACTUAL_EFFORT_S | EFFORT_ID |

RPT_FCT_TM_COMPLIANCE

This table stores time-sheet compliance information.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|---------------|----------|-----------|---|
| COMPLIANCE_ID | NOT NULL | NUMBER | System-generated identifier by the sequence RPT_FCT_TM_COMPLIANCE_S |
| RESOURCE_ID | NOT NULL | NUMBER | ID of the Resource who submitted this Timesheet. |
| TIME_SHEET_ID | NULLABLE | NUMBER | ID of the Timesheet represented by this row. |
| PERIOD_ID | NULLABLE | NUMBER | ID of the Period for which this Timesheet was submitted. |

| Column Name | Null? | Data Type | Description |
|---------------------|----------|-----------|---|
| FIRST_SUBMIT_OFFSET | NULLABLE | NUMBER | Difference (in days) between the date on which the time sheet was first submitted and the end date of the time-sheet period. A value of 0 (zero) indicates that it was submitted on the final day of the period. A negative value indicates that the time sheet was submitted before the end of the period. A positive value indicates that the time sheet was submitted after the end of the period. |
| FIRST_SUBMIT_DATE | NULLABLE | DATE | Date on which the time sheet was first submitted. |
| LAST_SUBMIT_OFFSET | NULLABLE | NUMBER | Difference (in days) between the date on which the time sheet was last submitted and the end date of the time sheet period. A value of 0 indicates that the time sheet was submitted on the final day of the period. A negative value indicates that the time sheet was submitted before the end of the period. A positive value indicates that the time sheet was submitted after the period ended. |
| LAST_SUBMIT_DATE | NULLABLE | DATE | Date on which the time sheet was last submitted. |
| SUBMIT_COUNT | NULLABLE | NUMBER | Number of times the resource submitted this time sheet. |
| FIRST_REWORK_OFFSET | NULLABLE | NUMBER | Difference (in days) between the date on which the time sheet was first moved to rework status and the end date of the time-sheet period. A value of 0 indicates that the time sheet was reworked on the final day of the period. A negative value indicates that the time sheet was reworked before the end of the period. A positive value indicates that the time sheet was reworked after the period ended. |
| FIRST_REWORK_DATE | NULLABLE | DATE | Date on which the time sheet was first reworked. |
| LAST_REWORK_OFFSET | NULLABLE | NUMBER | Difference (in days) between the date on which the time sheet was last moved to rework status and the end date of the time-sheet period. A value of 0 indicates that the time sheet was reworked on the final day of the period. A negative value indicates that the time sheet was reworked before the end of the period. A positive value indicates that the time sheet was reworked after the period ended. |

| Column Name | Null? | Data Type | Description |
|-----------------------|----------|-----------|---|
| LAST_REWORK_DATE | NULLABLE | DATE | Date on which the time sheet was last reworked. |
| REWORK_COUNT | NULLABLE | NUMBER | Number of times the resource reworked the time sheet. |
| FIRST_APPROVAL_OFFSET | NULLABLE | NUMBER | Difference (in days) between the date on which the time sheet was first approved and the end date of the time-sheet period. A value of 0 indicates that the time sheet was approved on the final day of the period. A negative value indicates that the time sheet was approved before the end of the period. A positive value indicates that the time sheet was approved after the period ended. |
| FIRST_APPROVAL_DATE | NULLABLE | DATE | Date on which the time sheet was first approved. |
| LAST_APPROVAL_OFFSET | NULLABLE | NUMBER | Difference (in days) between the date on which the time sheet was last approved and the end date of the time-sheet period. A value of 0 indicates that the time sheet was approved on the final day of the period. A negative value indicates that the time sheet was approved before the end of the period. A positive value indicates that the time sheet was approved after the period ended. |
| LAST_APPROVAL_DATE | NULLABLE | DATE | Date on which the time sheet was last approved. |
| APPROVAL_COUNT | NULLABLE | NUMBER | Number of times the time sheet was approved. |
| FROZEN_OFFSET | NULLABLE | NUMBER | Difference (in days) between the date on which the time sheet was frozen and the end date of the time-sheet period. A value of 0 indicates that the time sheet was frozen on the final day of the period. A negative value indicates that the time sheet was frozen before the end of the period. A positive value indicates that the time sheet was frozen after the period ended. |
| FROZEN_DATE | NULLABLE | DATE | Date on which the time sheet was frozen. |

| Column Name | Null? | Data Type | Description |
|---------------|----------|----------------|---|
| FROZEN_COUNT | NULLABLE | NUMBER | Number of times the time sheet was frozen. |
| CLOSED_OFFSET | NULLABLE | NUMBER | Difference (in days) between the date on which the time sheet was closed and the end date of the time-sheet period. A value of 0 indicates that the time sheet was closed on the final day of the period. A negative value indicates that the time sheet was closed before the end of the period. A positive value indicates that the time sheet was closed after the period ended. |
| CLOSED_DATE | NULLABLE | DATE | Date on which the time sheet was closed. |
| CLOSED_COUNT | NULLABLE | NUMBER | Number of times the time sheet was closed. |
| NLS_STATUS | NULLABLE | VARCHAR2 (150) | Time Sheet status including the missing status in NLS |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|--------------------------|------------|----------|---------------|
| RPT_FCT_TM_COMPLIANCE_B1 | NONUNIQUE | 1 | PERIOD_ID |
| RPT_FCT_TM_COMPLIANCE_B2 | NONUNIQUE | 1 | TIME_SHEET_ID |
| RPT_FCT_TM_COMPLIANCE_B3 | NONUNIQUE | 1 | RESOURCE_ID |
| SYS_C003624092 | UNIQUE | 1 | COMPLIANCE_ID |

Sequences

| Sequence Name | Sequence Type |
|-------------------------|---------------|
| RPT_FCT_TM_COMPLIANCE_S | COMPLIANCE_ID |

RPT_FCT_TM_COST

This table stores the Effort data for a Work Item on a Time Sheet Line.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|--------------------|----------|----------------|---|
| COST_ID | NOT NULL | NUMBER | System-generated identifier by the sequence RPT_FCT_TM_COST_S |
| TIME_SHEET_LINE_ID | NOT NULL | NUMBER | ID derived from the PPM TM_TIME_SHEET_LINES table. |
| TIME_SHEET_ID | NOT NULL | NUMBER | ID derived from the PPM TM_TIME_SHEETS table. |
| ACT_CAP_LABOR_BSE | NULLABLE | NUMBER | Actual Capitalized Cost. |
| ACT_OP_LABOR_BSE | NULLABLE | NUMBER | Actual Operational Cost. |
| CUR_BSE | NULLABLE | VARCHAR2 (255) | Base currency used for actual cost on time sheets. |
| ACTUAL_EFFORT | NULLABLE | NUMBER | Actual effort in hours on the time sheet line. |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|--------------------|------------|----------|--------------------|
| RPT_FCT_TM_COST_B1 | NONUNIQUE | 1 | TIME_SHEET_ID |
| RPT_FCT_TM_COST_B2 | NONUNIQUE | 1 | TIME_SHEET_LINE_ID |
| SYS_C003624099 | UNIQUE | 1 | COST_ID |

Sequences

| Sequence Name | Sequence Type |
|-------------------|---------------|
| RPT_FCT_TM_COST_S | COST_ID |

RPT_KNTA_USERS_DELETES

This table holds data during incremental ETL to determine which users need to be deleted. This is temporary table.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-------------|----------|-----------|--|
| USER_ID | NULLABLE | NUMBER | ID of the user derived from PPM KNTA_USERS table |

Indexes

This table uses no indexes.

Sequences

This table uses no sequences.

RPT_PARAMS

This table holds different parameters for the Full ETL and the incremental ETL to run successfully. For example, the db link name, start and end dates of the etl are all stored in this table.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-----------------|----------|----------------|---|
| PARAMETER_NAME | NOT NULL | VARCHAR2 (50) | A unique string value to represent the parameter name. |
| PARAMETER_VALUE | NULLABLE | VARCHAR2 (100) | String value for the corresponding parameter name. The strings can be converted to any data type as needed. For example, the start date will be converted to date type. |
| DESCRIPTION | NULLABLE | VARCHAR2 (500) | Description about this parameter. |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|----------------|------------|----------|----------------|
| SYS_C003623112 | UNIQUE | 1 | PARAMETER_NAME |

Sequences

This table uses no sequences.

RPT_PM_PROJECT_ENTRIES

This is an intermediate table to keep project identifiers for the fact calculation.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-------------|----------|-----------|-----------------|
| PROJECT_ID | NULLABLE | NUMBER | ID of a project |

Indexes

This table uses no indexes.

Sequences

This table uses no sequences.

RPT_PM_PROJECT_MANAGERS

RPT_PM_PROJECT_MANAGERS keeps information about manager(s) of a PPM projects.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-----------------|----------|-----------------|--|
| PROJECT_ID | NOT NULL | NUMBER | ID of a PPM project and it is mapped to project_id column of PM_PROJECTS table |
| MGR_USER_ID | NULLABLE | NUMBER | User ID of a project manager of a PPM project and it is related to user_id of KNTA_USERS table |
| ALL_PROJECT_MGR | NULLABLE | VARCHAR2 (4000) | It keeps all managers name of a PPM project. Names are separated by comma |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|---------------------|------------|----------|-------------|
| RPT_PM_PROJ_MGRS_N1 | NONUNIQUE | 1 | PROJECT_ID |
| RPT_PM_PROJ_MGRS_N2 | NONUNIQUE | 1 | MGR_USER_ID |

Sequences

This table uses no sequences.

RPT_PM_TASK_ENTRIES

This is an intermediate table to keep task identifiers for the fact calculation.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-------------|----------|-----------|--------------|
| TASK_ID | NULLABLE | NUMBER | ID of a task |

Indexes

This table uses no indexes.

Sequences

This table uses no sequences.

RPT_PPM_AGGR_WRK_DAYS_MV

This is a materialized view that aggregates the information in the KDRV_WORKING_DAYS with the help of PPM_FISCAL_PERIODS table to give the working hours for a given period.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|------------------|----------|-----------|---|
| CALENDAR_ID | NOT NULL | NUMBER | The calendar ID is derived from PPM's KDRV_WORKING_DAYS.CALENDAR_ID or KDRV_CALENDARS.CALENDAR_ID. |
| NUM_WRK_DAYS | NULLABLE | NUMBER | NUM_WRK_DAYS is the sum of the working days in this given period and for this calendar. |
| PERIOD_TYPE | NOT NULL | NUMBER | PERIOD_TYPE is a number to indicate, whether this period is a week, month, quarter, half year or year. |
| START_DATE | NOT NULL | DATE | START_DATE indicates the start of the period. |
| END_DATE | NOT NULL | DATE | END_DATE indicates the end of the period. |
| WORK_HRS_PER_DAY | NOT NULL | NUMBER | WORK_HRS_PER_DAY is the number of work hours in the day for this calendar. Derived from PPM's KDRV_CALENDAR_SETUPS table. |

Indexes

This table uses no indexes.

Sequences

This table uses no sequences.

RPT_RECREATE_INDEX

This table stores SQL statement for Index creation from memory for rebuilding the deleted indexes.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|--------------|----------|-----------------|--|
| INDEX_SQL_ID | NOT NULL | NUMBER | System-generated identifier derived from sequence RPT_RECREATE_INDEX_SEQ |
| SQL_STMT | NOT NULL | VARCHAR2 (4000) | SQL statement for Index creation from memory |

Indexes

This table uses no indexes.

Sequences

| Sequence Name | Sequence Type |
|------------------------|---------------|
| RPT_RECREATE_INDEX_SEQ | INDEX_SQL_ID |

RPT_RERUN_STATUS

This table stores records of the completion of tasks during initial data load in order that tasks already completed will not be run again if the process has to be restarted.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-----------------|----------|-----------|----------------------------|
| RERUN_STATUS_ID | NOT NULL | NUMBER | Uniquely identify a record |

| Column Name | Null? | Data Type | Description |
|-----------------|----------|----------------|---|
| COMPLETION_DATE | NOT NULL | DATE | When the task was completed |
| FILE_NAME | NULLABLE | VARCHAR2 (512) | The file containing the code for the task |
| ACTION_NAME | NULLABLE | VARCHAR2 (512) | Name of the completed action |
| RUN_ID | NULLABLE | NUMBER | The run in which this action completed, taken from the RPT_RERUN_RUN_S sequence |
| VERSION | NULLABLE | VARCHAR2 (100) | Version of PPM this load was run against |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|----------------|------------|----------|-----------------|
| SYS_C007867316 | UNIQUE | 1 | RERUN_STATUS_ID |
| UNIQUE_ACTION | UNIQUE | 1 | FILE_NAME |
| UNIQUE_ACTION | UNIQUE | 2 | ACTION_NAME |
| UNIQUE_ACTION | UNIQUE | 3 | VERSION |

Sequences

| Sequence Name | Sequence Type |
|--------------------|-----------------|
| RPT_RERUN_RUN_S | RUN_ID |
| RPT_RERUN_STATUS_S | RERUN_STATUS_ID |

RPT_RM_FACT_CNTL

This is a single column table, that is used during the RM (Resource Management) incremental etl. It stores the name of the table that currently is being processed for the changes occurred in PPM. Reading this table the switchable views builds the view to get data from the original PPM table or only the changes made on the PPM table via CDC table. When there is nothing to process, it is populated with a dummy number.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-------------|----------|-----------|--|
| CDC | NULLABLE | CHAR | Contains the table name that is currently being processed from the subscribed list of PPM tables for change capture. |

Indexes

This table uses no indexes.

Sequences

This table uses no sequences.

RPT_RM_RSC_RP_ENTRIES

This table stores some temporary information needed during RM incremental etl processing. When the delete events happen on PPM RSC_RP_DISTRIBUTION_ENTIRES table, during the incremental etl we track RESOURCE_POOL_ID and resource_id in this table, and later this data is used to delete the combination of the same entries from aggregated fact tables.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|------------------|----------|-----------|--|
| RESOURCE_POOL_ID | NULLABLE | NUMBER | The RESOURCE_POOL_ID with the combination of RESOURCE_ID that are deleted. |
| RESOURCE_ID | NULLABLE | NUMBER | The RESOURCE_ID with the combination of RESOURCE_POOL_ID that are deleted. |

Indexes

This table uses no indexes.

Sequences

This table uses no sequences.

RPT_RSC_RPDE_DELETES

This is a staging table used in incremental ETL process for storing the delete candidates which will be deleted from the RPT_FCT_RM_RESOURCE_CAPACITY fact table due to changes occurred in the RSC_RP_DISTRIBUTION_ENTRIES table.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-------------|----------|-----------|---|
| CAPACITY_ID | NULLABLE | NUMBER | CAPACITY_ID derived from RPT_FCT_RM_RESOURCE_CAPACITY table |

| Column Name | Null? | Data Type | Description |
|------------------|----------|-----------|--|
| RESOURCE_POOL_ID | NULLABLE | NUMBER | ID of the resource pool derived from RSC_RP_DISTRIBUTION_ENTRIES table |
| RESOURCE_ID | NULLABLE | NUMBER | ID of the resource derived from RSC_RP_DISTRIBUTION_ENTRIES table |

Indexes

This table uses no indexes.

Sequences

This table uses no sequences.

RPT_TRACE_DETAILS

This table stores the file, line, and function that need to perform SQL level trace during full ETL or incremental ETL. Enable trace log for the SQLs you noted down by adding the file name, line number, and function name of the rows to the RPT_TRACE_DETAILS table.

If you add file name and set line number to -1, the SQL trace log stays open for the entire package body file.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-------------|----------|-----------|---|
| FILE_NAME | NOT NULL | NUMBER | PLSQL package body file name |
| LINE_NO | NULLABLE | NUMBER | Line number in package body source file |
| FUNC_NAME | NULLABLE | NUMBER | Function name of the package |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|----------------------|------------|----------|-------------|
| RPT_TRACE_DETAILS_N1 | NONUNIQUE | 1 | FILE_NAME |
| RPT_TRACE_DETAILS_N2 | NONUNIQUE | 1 | LINE_NO |
| RPT_TRACE_DETAILS_N3 | NONUNIQUE | 1 | FUNC_NAME |

Sequences

This table does not use any sequences.

RPT_UPGRADE_PARAMS

This table holds data for upgrade process. During upgrade, it checks this table to see the status of pre-upgrade stage.

Foreign Keys

This table does not have any foreign keys.

Column Descriptions

| Column Name | Null? | Data Type | Description |
|-----------------|----------|---------------|------------------------------------|
| PARAMETER_NAME | NOT NULL | VARCHAR2(50) | Unique name a parameter |
| PARAMETER_VALUE | NULLABLE | VARCHAR2(100) | Value of the parameter |
| DESCRIPTION | NULLABLE | VARCHAR2(500) | Brief description of the parameter |

Indexes

| Index Name | Index Type | Sequence | Column Name |
|----------------|------------|----------|----------------|
| SYS_C003670537 | UNIQUE | 1 | PARAMETER_NAME |

Sequences

This table uses no sequences.

Send Documentation Feedback

If you have comments about this document, you can [contact the documentation team](#) by email. If an email client is configured on this system, click the link above and an email window opens with the following information in the subject line:

Feedback on Operational Reporting Administrator's Guide (Project and Portfolio Management Center Content Pack 2.0)

Just add your feedback to the email and click send.

If no email client is available, copy the information above to a new message in a web mail client, and send your feedback to HPSW-BTO-PPM-SHIE@hp.com.

We appreciate your feedback!