

HP Project and Portfolio Management Center

Software Version: 9.20

HP Deployment Management Extension for Oracle Technology Guide

Document Release Date: March 2013 (Second Edition)

Software Release Date: February 2013



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 Notices

© Copyright 1997-2013 Hewlett-Packard Development Company, L.P.

Trademark Notices

Adobe® is a trademark of Adobe Systems Incorporated.

Intel®, Intel® Itanium®, Intel® Xeon®, and Pentium® are trademarks of Intel Corporation in the U.S. and other countries.

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

Oracle and Java are registered trademarks of Oracle and/or its affiliates.

UNIX® is a registered trademark of The Open Group.

Documentation Updates

This manual's title page 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:

h20230.www2.hp.com/selfsolve/manuals

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

The following table indicates changes made to this document.

Publication Date	Summary of Changes
March 2013 (9.20, Second Edition)	■ Corrected version errors.

Support

You can visit the HP Software Support Web site at:

hp.com/go/hpsoftwaresupport

HP Software Support Online provides an efficient way to access interactive technical support tools. As a valued support customer, you can benefit by using the support 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 find more information about access levels, go to:

h20230.www2.hp.com/new_access_levels.jsp

To register for an HP Passport ID, go to:

h20229.www2.hp.com/passport-registration.html

Contents

1	Getting Started	7
	Introduction to the Extension for Oracle Technology	7
	What's New and What's Changed in Version 9.20	8
	About This Document	8
	Related Information	9
	Prerequisite Knowledge and Experience	10
2	Installing or Upgrading the Extension	11
	Overview of Installation and Upgrade	11
	System Requirements	11
	General Upgrade Impacts and Guidelines	12
	Specific Upgrade Impacts for Version 9.20	13
	Preparing for Installation or Upgrade	13
	General Preparations for Installation or Upgrade	13
	Performing Backup and Restarting the PPM Server in Restricted Mode	14
	Installation Procedure	15
	Run the Installation Script and Check the Logs	15
	Logs Generated During Installation	16
	Verify the Installation	17
	Restart the PPM Server in Normal Mode	18
	Post-Installation Procedures	18
3	Extension Object Types	19
	Overview of Object Types	19
	Reference Object Types	20
	List of Object Types	21
	Forms 4.5 Object Type	22
	Forms 6.0 Object Type	23

Forms 10G Object Type	25
Reports 2.0 Object Type	26
Reports 2.5 Object Type	28
Reports 6.0 Object Type	29
Reports 10G Object Type	30
SQL Loader File Object Type.....	32
SQL Loader80 File Object Type	34
SQL Script Object Type	37
Index	39

1 Getting Started

Introduction to the Extension for Oracle Technology

HP Deployment Management Extension for Oracle Technology (usually referred to hereafter as “the Extension”) helps organizations to automate deployment management in application environments built using Oracle® tools such as SQL*Plus, PL/SQL, Oracle Forms, and Oracle Reports.

The Extension enhances the functionality of the HP Deployment Management application in HP Project and Portfolio Management Center (PPM Center) by providing predefined object types and commands specifically related to development using Oracle tools.

After PPM Center (the “base” software) has been installed at or upgraded to version 9.20, you can install the Extension or upgrade it to version 9.20 on the same system.

[Chapter 2, *Installing or Upgrading the Extension*, on page 11](#) describes system requirements for installing the Extension or upgrading it to version 9.20, and the impacts of upgrading the Extension to version 9.20.

What's New and What's Changed in Version 9.20

Extension version 9.20 makes no functional changes compared to version 9.10. Upgrading to version 9.20 does not affect existing Extension functionality.

After installing PPM Center at version 9.20, you can install Extension version 9.20 for the first time, or you can upgrade the Extension from version 9.10.

For information about changes to PPM Center in version 9.20, see the *What's New and What's Changed* document.

PPM Center supports having multiple languages on the same instance, including translations of specific PPM Center entities and interfaces. HP does not provide translations of any Extension entities or interfaces with the Extension or any language pack.



However, like any customer-defined entity, Extension entities such as request types can be translated by using the `kExportAttributes.sh` and `kImportAttributes.sh` scripts as part of a translation process. For more information, see the *Multilingual User Interface Guide*.

[Chapter 2, *Installing or Upgrading the Extension*, on page 11](#) describes general impacts of upgrading the Extension from version 9.10 to version 9.20.

About This Document

This document is intended for the following audiences:

- Oracle Applications developers building applications using Oracle tools such as SQL*Plus, PL/SQL, Oracle Forms, and Oracle Reports
- Database or application administrators responsible for installing or maintaining the toolset, or controlling changes in one or more application environments built on the Oracle toolset
- Database or application administrators responsible for maintaining access and security for, or supporting use of, HP Deployment Management Extension for Oracle Technology

This guide provides information about installing or upgrading to version 9.20 of HP Deployment Management Extension for Oracle Technology, and this guide provides conceptual, procedural, and reference information about the product.

This guide is organized as follows:

- [Chapter 1, *Getting Started*, on page 7](#) (this chapter) provides an introduction to the Extension, the features and changes introduced in version 9.20, information about the intended audiences for this document, related information, and prerequisite knowledge.
- [Chapter 2, *Installing or Upgrading the Extension*, on page 11](#) provides overview and detailed information about installing or upgrading the Extension, including information about the impacts of upgrading the Extension.
- [Chapter 3, *Extension Object Types*, on page 19](#) provides information about the Extension's object types.

Related Information

The following documents for PPM Center version 9.20 also include information useful in managing HP Deployment Management Extension for Oracle Technology:

- *Installation and Administration Guide*
- *System Requirements and Compatibility Matrix*
- *Security Model Guide and Reference*
- *What's New and What's Changed*
- *HP Deployment Management Configuration Guide*
- *HP Deployment Management User's Guide*

If you are upgrading the Extension from version 9.10, details about that release are available in the *HP Deployment Management Extension for Oracle Technology Guide* for version 9.10.

The following additional Extension and Migrator documentation for Oracle environments might be of particular interest:

- *HP Object Migrator Guide*
- *HP GL Migrator Guide*
- *HP Deployment Management Extension for Oracle E-Business Suite Guide*

Prerequisite Knowledge and Experience

To install, upgrade, configure, maintain, or use the Extension, you need to understand the following:

- Deployment management
- Environments
- Software deployment
- Packages
- PPM Dashboard pages and portlets
- PPM Workbench
- Object types
- Workflows and workflow steps
- Tokens
- PPM Center entities installed by the Extension

In addition, you must have practical experience in the following:

- Installing, upgrading, configuring, and using PPM Center, if you are responsible for configuring the Extension
- Installing, configuring, and using Oracle tools

2 Installing or Upgrading the Extension

Overview of Installation and Upgrade

This section describes system requirements that must be met before installing or upgrading the Extension, and this section describes upgrade impacts.

System Requirements

Before you install the Extension or upgrade it to version 9.20, PPM Center version 9.20 must be installed. PPM Center and the Extension are installed on the same system and have the same system requirements.

For information about installing PPM Center version 9.20, see the *Installation and Administration Guide*. For information about upgrading PPM Center to version 9.20, see the *Upgrade Guide* for version 9.20.

For information about version levels and other general system requirements, see the *System Requirements and Compatibility Matrix*, available at the HP Software Product Manuals Web site:

h20230.www2.hp.com/selfsolve/manuals

General Upgrade Impacts and Guidelines

Each new version or service pack of the Extension provides the following types of entities, which the Extension upgrade process installs or does not install in the PPM Center instance, as described:

- **Reference entities.** The names of reference entities start with (REFERENCE).

You cannot edit reference entities. However, you can copy and rename them and then edit the copies as non-reference entities, as appropriate for your environment.

The upgrade process deletes all of the Extension's existing (previously installed) reference entities, such as its object types, and then the process installs a new set of reference entities for the new version. The new set of reference entities can be identical to the previous set or, to support changes in functionality, the new set can have new, deleted, renamed, or modified reference entities.

- **Non-reference entities.** The non-reference entities and corresponding reference entities that the version or service pack provides are identical, except that the names of the reference entities start with (REFERENCE).

You can edit (or copy and edit) non-reference entities, as appropriate for your environment. The currently installed version of the Extension might include non-reference entities that have been edited (customized).

To preserve all existing customizations, the upgrade process compares the names of the non-reference entities delivered with the new version to the names of existing non-reference entities in the instance, and if the process finds an existing non-reference entity with the same name, the process does *not* overwrite or modify that entity in the instance. If the new version introduces reference entities with new names along with non-reference copies of those entities, then the upgrade process adds both the reference and non-reference copies.

In general, if an upgrade changes an Extension's reference entities, you must evaluate how those changes should affect the associated, previously customized non-reference entities. After the upgrade is performed, you can revise those non-reference entities or create new ones as necessary.

Specific Upgrade Impacts for Version 9.20

Upgrading to version 9.20 does not affect existing Extension functionality.

Preparing for Installation or Upgrade

Prepare for installation or upgrade of the Extension as described in the following sections.



During installation or upgrade, the PPM Server must run in restricted mode.

General Preparations for Installation or Upgrade

To prepare for installation or upgrade of the Extension:

1. Obtain the Extension software.
2. Collect the following information, which you will need to supply during the installation procedure:
 - The logon username and password for the Extension's server (the same server on which PPM Center is or will be installed). The username (typically "admin") must belong to a security group that has the following access grants:
 - Sys Admin: Migrate PPM Objects
 - Sys Admin: Server Administrator
 - The database password for the server's schema.
3. Log on to the PPM Server.
4. Verify that the system requirements have been met. See *System Requirements* on page 11.

5. Save the Extension installation file (`ppm-920-OracleTech.jar`) to one of the following directories:
 - `<PPM_Home>` (the recommended location).
`<PPM_Home>` represents the path where your PPM Center instance is installed. For example: `xyzserver/E/PPMServer`.
 - A subdirectory of `<PPM_Home>`. If a subdirectory of `<PPM_Home>` is specified in the `ITG_DEPLOYMENT_HOME` environment variable, the installation script finds and uses the installation file in that subdirectory. If you want to save the installation file to a subdirectory of `<PPM_Home>`, make sure the value of the `ITG_DEPLOYMENT_HOME` environment variable is `<PPM_Home>` followed by that subdirectory, for example, `<PPM_Home>/Extension`.
 - Any directory (with, optionally, any subdirectories) you choose, for example, `dirA/sub1/sub2`.

Performing Backup and Restarting the PPM Server in Restricted Mode

The steps in this section are recommended but not required.



For more information about the steps in this procedure, see the *Installation and Administration Guide*.

For a new installation or an upgrade, do the following:

1. Back up the database and file system for the PPM Server.
2. To stop the PPM Server and restart it in restricted mode:
 - a. Stop the PPM Server.
 - b. Run the following script:

```
sh ./setServerMode.sh RESTRICTED
```
 - c. Start the PPM Server.

Installation Procedure

Perform the procedures in the following sections to install the Extension.

Run the Installation Script and Check the Logs

To run the installation script to install the Extension, and to check the logs:

1. Be sure you have completed all the steps in *Preparing for Installation or Upgrade* on page 13.

In particular, be sure the PPM Server is running in restricted mode. See *Performing Backup and Restarting the PPM Server in Restricted Mode* on page 14.

2. On the PPM Server, navigate to the `bin` subdirectory of the `<PPM_Home>` directory (or other directory as described in [step 5 on page 14](#)).
3. Start the installation or upgrade. In [step 5 on page 14](#), if you saved the installation file to `<PPM_Home>`, which is the recommended directory, or to a subdirectory of `<PPM_Home>`, run the following script:

```
sh ./kDeploy.sh -i OracleTech
```

However, if you saved the installation file to a different directory, see the example in [step 5 on page 14](#) and specify that directory in the script command, as in the following example:

```
sh ./kDeploy.sh -i OracleTech -D dirA/sub1/sub2
```

4. Follow the script's on-screen prompts to complete the installation. Prompts can include the database password for the schema and the logon name and password for the server.

Files are installed in various subdirectories under `<PPM_Home>`. Data is also placed in the database. When the installation script is complete, the following message appears:

```
Deployment OracleTech has been successfully installed.
```

5. Use a Web browser to check the installation summary report, which is located at:

```
<PPM_Home>/logs/deploy/920/OracleTech/<Log_x>/  
installLog.html
```

where *<Log_x>* is initially a random number generated by `kDeploy.sh` during installation. The number increments by one each time the installation script is run, so the installation summary report for the most recent run is in the directory with the highest number.

The installation summary report lists all the entities that are installed as part of the Extension installation process. Each entity that was installed correctly is marked as Complete. If there is an error for a particular entity, the report contains a direct link to another log file (HTML page) with additional information.

If necessary, correct any errors and repeat the installation process.

Installation of the Extension generates the logs that are described in *Logs Generated During Installation*, depending on the installation options.

Logs Generated During Installation

Depending on the installation options that were chosen, the logs listed and described in *Table 2-1* can be generated during installation and saved in the following directory:

```
<PPM_Home>/logs/deploy/920/OracleTech
```

The log number (*<#####>*) shown in *Table 2-1* is a random number (generated by `kDeploy.sh`) that makes each log file name unique.

Table 2-1. Logs generated during installation (page 1 of 2)

File Name	Description
ddlDriver.<#####>.log	Contains information about data model changes made during installation
jarxf.<#####>.log	Contains information from the procedure that unpacks the .jar file

Table 2-1. Logs generated during installation (page 2 of 2)

File Name	Description
packageDriver.<#####>.log	Contains information about the installation of database code; for example, reports
postXMLDriver.<#####>.log	Contains information about the application of SQL scripts required after the installation of HP-supplied data
preXMLDriver.<#####>.log	Contains information about the application of SQL scripts required before the installation of HP-supplied data, such as the definition for HP Deployment Management Extension for Oracle Technology

Verify the Installation

HP strongly recommends that you verify correct installation. To verify that Extension version 9.20 for Oracle Technology is listed among the installed Extensions, navigate to the <PPM_Home>/bin directory and run the following script:

```
sh ./kDeploy.sh -l
```

where the last character in the command is the lowercase letter “l.”

The name OracleTech should appear in the list of installed Extensions.

For example, if both HP Deployment Management Extension for Oracle Technology and HP Deployment Management Extension for Oracle E-Business Suite are now installed at version 9.20, the following table entries are displayed:

Deployment	Version	Deployed	Description
OracleTech	920	<date and time>	OracleTech Extension
OracleApps	920	<date and time>	Oracle Apps Extension

Restart the PPM Server in Normal Mode



For more information about the steps in this procedure, see the *Installation and Administration Guide*.

After you have completed all installation or upgrade procedures, if you previously restarted the PPM Server in restricted mode, to stop and restart it in normal mode:

1. Stop the PPM Server.
2. Run the following script:

```
sh ./setServerMode.sh NORMAL
```

3. Start the PPM Server.

Post-Installation Procedures

After you have finished installing HP Deployment Management Extension for Oracle Technology, you must do the following:

- Review the object types.
- Define environments for each application environment that will be a source or destination of migration, for example:
 - Host connection information
 - Database connection information
- If required, customize the logic in the various installed entities.



After the Extension has been running successfully for a substantial period of time, you can optionally delete all of the installation files. However, HP recommends that you retain (or copy) the log files.

3 Extension Object Types

Overview of Object Types

This chapter provides reference information about the Oracle Technology-specific object types provided in the Extension. These object types are listed and defined in *Table 3-1 on page 21*.

Migration and compilation of object types are driven by commands included within the object types. For more information about commands in the PPM Center environment, see the *Commands, Tokens, and Validations Guide and Reference*. For more information about using object types in packages, see the *HP Deployment Management User's Guide*.

You can view or modify an object type as follows:

1. Log on to PPM Center.
2. From the menu bar, select **Open > Administration > Open Workbench**.
The PPM Workbench opens.
3. From the shortcut bar, select **Deployment Mgmt > Object Types**.
The Object Type Workbench opens.
4. (Optional) Select **Oracle Technology** in the **Extension** field on the **Query** tab.
5. In the Object Type Workbench, click **List**.
6. Select the object type of interest and click **Open**.
7. Edit the object type. (On the **Fields** tab, the list of fields in the **Prompts** column is alphabetized.)

Subsequent figures in this chapter show windows you can use to revise object types in conjunction with adding package lines. You can access these windows as follows:

1. Log on to PPM Center.
2. From the menu bar, select **Open > Administration > Open Workbench**.

The PPM Workbench opens.

3. From the shortcut bar, select **Deployment Mgmt > Packages**.

The Package Workbench opens.

4. Add a new or open an existing package, as necessary.
5. Select a workflow.
6. Add a line.
7. Select the object type of interest.

Reference Object Types

Reference object types cannot be edited, but you can copy and rename them and edit the copies to meet your needs. You can also use existing non-reference object types as is or configure them further to meet your needs.

List of Object Types

Table 3-1 lists and defines the object types included in the Extension. Each is described in subsequent sections.

Table 3-1. Object types included in the Extension

Object Type	Description
Forms 4.5	Moves Oracle Forms 4.5 files from one instance to another
Forms 6.0	Moves Oracle Forms 6.0 files from one instance to another
Forms 10G	Moves Oracle Forms 10G files from one instance to another
Reports 2.0	Moves Oracle Reports 2.0 files from one instance to another
Reports 2.5	Moves Oracle Reports 2.5 files from one instance to another
Reports 6.0	Moves Oracle Reports 6.0 files from one instance to another
Reports 10G	Moves Oracle Reports 10G files from one instance to another
SQL Loader File	Migrates the SQL Loader Control file from one instance to another and optionally runs SQL Loader
SQL Loader80 File	Migrates the SQL Loader control files from one instance to another and optionally runs SQL Loader against an Oracle 8 instance
SQL Script	Moves SQL and PL/SQL scripts from one instance to another and optionally executes the scripts at the destination

Forms 4.5 Object Type

The Forms 4.5 object type moves Oracle Forms 4.5 files from one instance to another. If the extension of the file being migrated is `fmb`, then the file is parsed at the source to generate a file with an `fmb` extension. Based on the `fmb` file, a file with an `fmw` extension is generated. The object type connects to the Oracle Account defined for the application (or on the **Host** tab, if the application value is null) in the destination environment to generate the `fmw` and `fmb` files. The `fmw` file is then moved to the destination instance.

You can modify this object type to include steps to interact with your version control system.

Figure 3-1 shows the default screen when adding a package line that uses the Forms 4.5 object type. *Table 3-2* provides field descriptions for the object type.

Figure 3-1. Forms 4.5 object type sample data

The screenshot shows a dialog box titled "Add Line" with a blue header. Below the header is a section titled "Object Type Information" containing three fields: "Object Type" (Forms 4.5), "Sequence" (1), and "Application Code" (None). Below this is a tabbed interface with "Parameters" and "User Data" tabs. The "Parameters" tab is active and contains three fields: "Sub Path" (forms/US), "Base File Name" (empty), and "File Extension" (fmb). At the bottom of the dialog are four buttons: "Clear", "OK", "Add", and "Cancel". A status bar at the very bottom of the dialog displays the text "Forms 4.5' parameters loaded."

Table 3-2. Forms 4.5 object type field descriptions

Field Name (*Required)	Description
Sub Path	Subpath (relative to base path) to the Forms 4.5 files
*Base File Name	Name of the file to be migrated
*File Extension	File extension of the file to be migrated— <code>fmt</code> , <code>fmb</code> , or <code>fmx</code>
File Location (Hidden by default)	Client or Server
File Type (Hidden by default)	ASCII or binary

Forms 6.0 Object Type

The Forms 6.0 object type moves Oracle Forms 6.0 files from one instance to another. If the extension of the file being migrated is `fmt`, then the file is parsed at the source to generate a file with an `fmb` extension. Based on the `fmb` file, a file with an `fmx` extension is generated. The object type connects to the Oracle Account defined for the application (or on the **Host** tab, if the application value is null) in the destination environment to generate the `fmx` and `fmb` files. The `fmx` file is then moved to the destination instance.

You can modify this object type to include steps to interact with your version control system.

Figure 3-2 shows the default screen when adding a package line that uses the Forms 6.0 object type. *Table 3-3* provides field descriptions for the object type.

Figure 3-2. Forms 6.0 object type sample data

The screenshot shows a dialog box titled "Add Line" with a blue title bar. It contains the following fields and controls:

- Object Type Information:**
 - Object Type: Forms 6.0 (text field)
 - Sequence: 1 (text field)
 - Application Code: None (dropdown menu)
- Parameters:**
 - Sub Path: forms/US (text field with browse button)
 - Base File Name: (empty text field with browse button)
 - File Extension: fnt (dropdown menu)
- Buttons:** Clear, OK, Add, Cancel
- Status Bar:** 'Forms 6.0' parameters loaded.

Table 3-3. Forms 6.0 object type field descriptions

Field Name (*Required)	Description
Sub Path	Subpath (relative to base path) to the Forms 6.0 files
*Base File Name	Name of the file to be migrated
*File Extension	File extension of the file to be migrated—fnt, fmb, or fmx
File Location (Hidden by default)	Client or Server
File Type (Hidden by default)	ASCII or binary

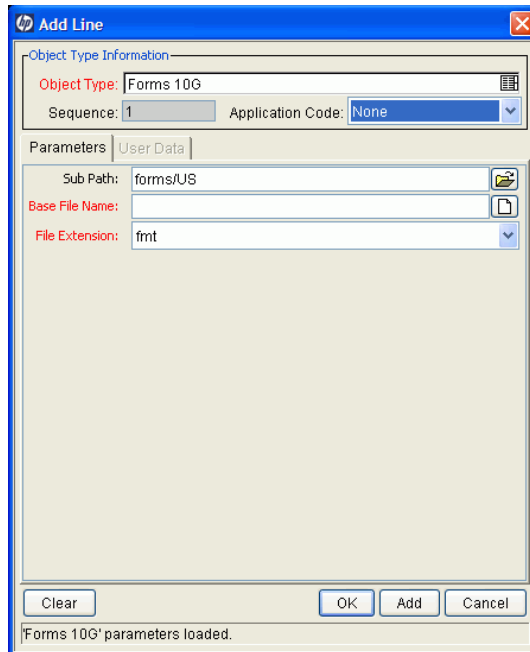
Forms 10G Object Type

The Forms 10G object type moves Oracle Forms 10G files from one instance to another. If the extension of the file being migrated is `fmt`, then the file is parsed at the source to generate a file with an `fmb` extension. Based on the `fmb` file, a file with an `fmx` extension is generated. The object type connects to the Oracle Account defined for the application (or on the **Host** tab, if the application value is null) in the destination environment to generate the `fmx` and `fmb` files. The `fmx` file is then moved to the destination instance.

You can modify this object type to include steps to interact with your version control system.

Figure 3-3 shows the default screen when adding a package line that uses the Forms 10G object type. *Table 3-4* provides field descriptions for the object type.

Figure 3-3. Forms 10G object type sample data



The screenshot shows a dialog box titled "Add Line" with a close button in the top right corner. The dialog is divided into two main sections: "Object Type Information" and "Parameters".

- Object Type Information:**
 - Object Type:** Forms 10G
 - Sequence:** 1
 - Application Code:** None
- Parameters:**
 - Sub Path:** forms/US
 - Base File Name:** (empty)
 - File Extension:** f_{mt}
- User Data:** (empty)

At the bottom of the dialog, there are four buttons: "Clear", "OK", "Add", and "Cancel". A status bar at the very bottom of the dialog reads "Forms 10G' parameters loaded."

Table 3-4. Forms 10G object type field descriptions

Field Name (*Required)	Description
Sub Path	Subpath (relative to base path) to the Forms 6.0 files
*Base File Name	Name of the file to be migrated
*File Extension	File extension of the file to be migrated— <code>fmt</code> , <code>fmb</code> , or <code>fmX</code>
File Location (Hidden by default)	Client or Server
File Type (Hidden by default)	ASCII or binary

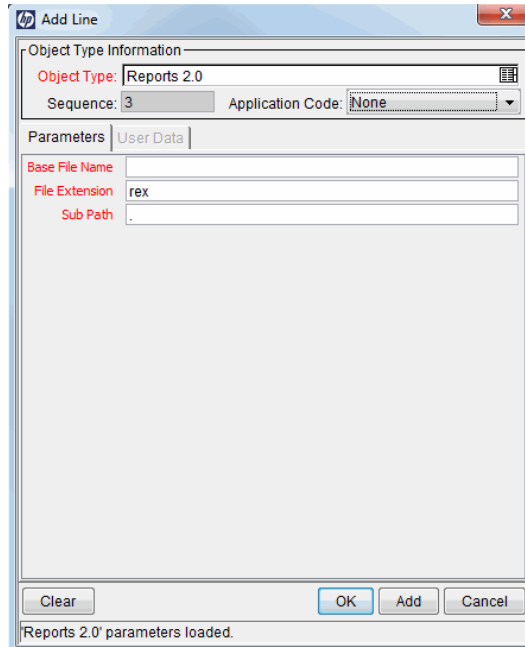
Reports 2.0 Object Type

The Reports 2.0 object type moves Oracle Reports 2.0 files from one instance to another. If a `rex` file is being moved, then an `rdf` file is generated at the destination. This migrates the report to the Oracle Account defined as the destination in the **Host** tab of the PPM Center Environment Workbench.

You can modify this object type to include steps to interact with your version control system.

Figure 3-4 shows the default screen when adding a package line that uses the Reports 2.0 object type. *Table 3-5* provides field descriptions for the object type.

Figure 3-4. Reports 2.0 object type sample data



The screenshot shows a software dialog box titled "Add Line". It contains a section for "Object Type Information" with a dropdown menu for "Object Type" set to "Reports 2.0", a text field for "Sequence" containing "3", and a dropdown for "Application Code" set to "None". Below this is a "Parameters" section with a "User Data" tab. It includes three input fields: "Base File Name" (empty), "File Extension" (containing "rex"), and "Sub Path" (containing "."). At the bottom of the dialog are "Clear", "OK", "Add", and "Cancel" buttons. A status bar at the very bottom indicates "Reports 2.0 parameters loaded."

Table 3-5. Reports 2.0 object type field descriptions

Field Name (*Required)	Description
*Base File Name	Name of the file to be migrated
*File Extension	File extension of the file to be migrated— <code>rex</code> or <code>rdf</code>
*Sub Path	Subpath (relative to base path) to the Reports 2.0 files

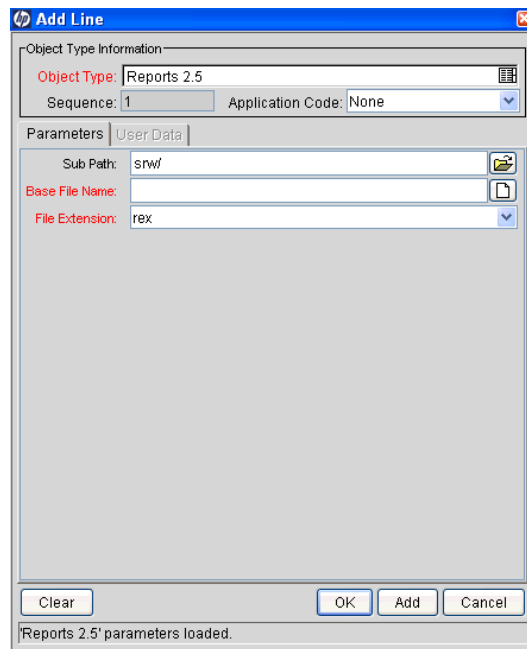
Reports 2.5 Object Type

The Reports 2.5 object type moves Oracle Reports 2.5 files from one instance to another. If a `rex` file is being moved, then an `rdf` file is generated at the destination. This migrates the report to the Oracle Account defined as the destination in the **Host** tab of the PPM Center Environment Workbench.

You can modify this object type to include steps to interact with your version control system.

Figure 3-5 shows the default screen when adding a package line that uses the Reports 2.5 object type. *Table 3-6* provides field descriptions for the object type.

Figure 3-5. Reports 2.5 object type sample data



The screenshot shows a dialog box titled "Add Line" with a close button in the top right corner. The dialog is divided into two main sections: "Object Type Information" and "Parameters".

Object Type Information:

- Object Type:** Reports 2.5 (text field)
- Sequence:** 1 (text field)
- Application Code:** None (dropdown menu)

Parameters: (tabbed view, currently showing "User Data")

- Sub Path:** srw/ (text field with folder icon)
- Base File Name:** (text field with document icon)
- File Extension:** rex (dropdown menu)

At the bottom of the dialog, there are four buttons: "Clear", "OK", "Add", and "Cancel". A status bar at the very bottom of the dialog displays the message: "Reports 2.5' parameters loaded."

Table 3-6. Reports 2.5 object type field descriptions

Field Name (*Required)	Description
Sub Path	Subpath (relative to base path) to the Reports 2.5 files
*Base File Name	Name of the file to be migrated
*File Extension	File extension of the file to be migrated— <code>rex</code> or <code>rdf</code>

Reports 6.0 Object Type

The Reports 6.0 object type moves Oracle Reports 6.0 files from one instance to another. If a `rex` file is being moved, then an `rdf` file is generated at the destination. This migrates the report to the Oracle Account defined as the destination in the **Host** tab of the PPM Center Environment Workbench.

You can modify this object type to include steps to interact with your version control system.

Figure 3-6 shows the default screen when adding a package line that uses the Reports 6.0 object type. *Table 3-7* provides field descriptions for the object type.

Figure 3-6. Reports 6.0 object type sample data

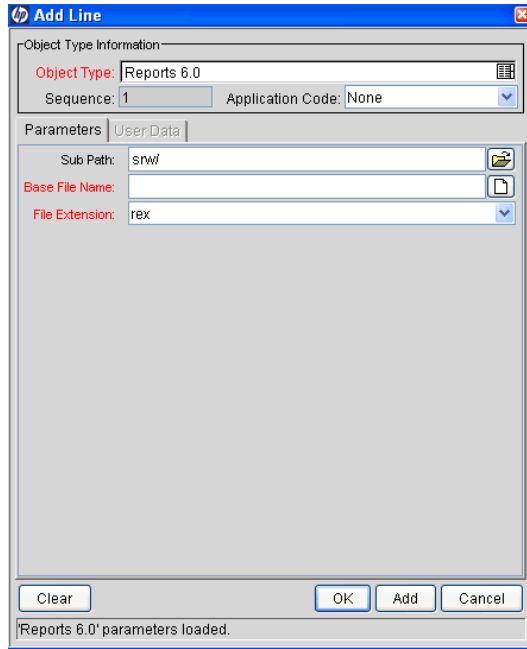


Table 3-7. Reports 6.0 object type field descriptions

Field Name (*Required)	Description
Sub Path	Subpath (relative to base path) to the Reports 6.0 files
*Base File Name	Name of the file to be migrated
*File Extension	File extension of the file to be migrated— <code>rex</code> or <code>rdf</code>

Reports 10G Object Type

The Reports 10G object type moves Oracle Reports 10G files from one instance to another. If a `rex` file is being moved, then an `rdf` file is generated at the destination. This migrates the report to the Oracle Account defined as the destination in the **Host** tab of the PPM Center Environment Workbench.

You can modify this object type to include steps to interact with your version control system.

Figure 3-7 shows the default screen when adding a package line that uses the Reports 10G object type. *Table 3-8* provides field descriptions for the object type.

Figure 3-7. Reports 10G object type sample data

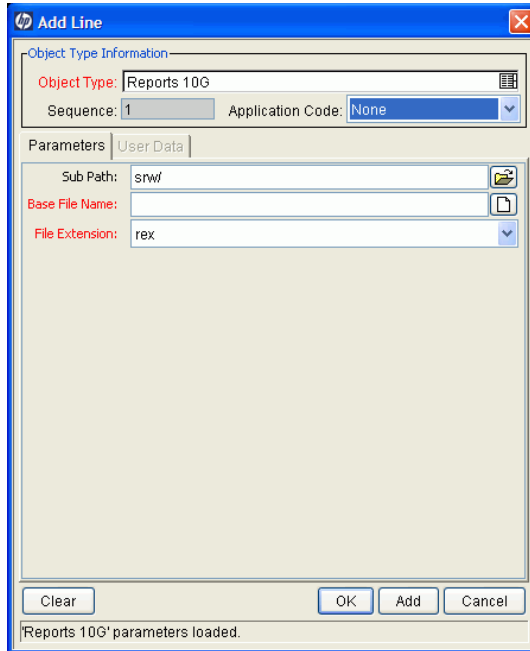


Table 3-8. Reports 10G object type field descriptions

Field Name (*Required)	Description
Sub Path	Subpath (relative to base path) to the Reports 10G files
*Base File Name	Name of the file to be migrated
*File Extension	File extension of the file to be migrated— <i>rex</i> or <i>rdf</i>

SQL Loader File Object Type

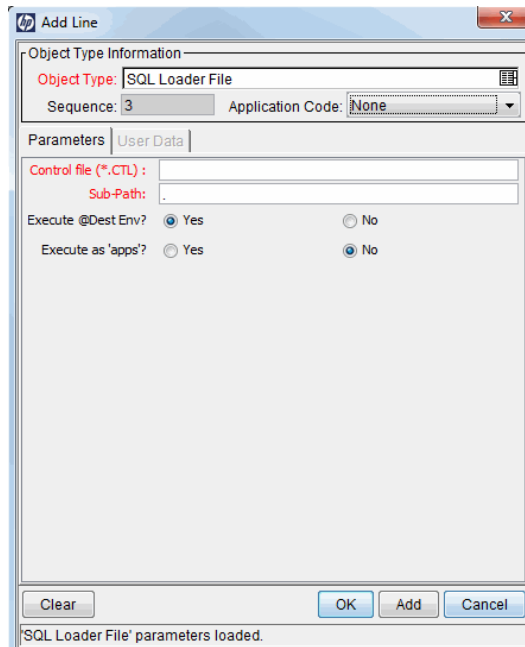
The SQL Loader File object type migrates the Oracle SQL Loader control file from one Oracle instance to another. The object type then optionally runs SQL Loader to load the data into an Oracle database at the destination instance.

Figure 3-8 shows the default screen when adding a package line that uses the SQL Loader File object type. *Table 3-9* provides field descriptions for the object type.



The SQL Loader File object type does not, by default, use any of the fields in *Table 3-9* that are not required. These fields are provided to allow users to customize and add to the SQL Loader call within the command steps.

Figure 3-8. SQL Loader File object type sample data



The screenshot shows a dialog box titled "Add Line" with a close button (X) in the top right corner. The dialog is divided into two main sections: "Object Type Information" and "Parameters".

Object Type Information:

- Object Type:** SQL Loader File (with a help icon)
- Sequence:** 3
- Application Code:** None (dropdown menu)

Parameters: (Selected tab)

- Control file (*.CTL):** (empty text field)
- Sub-Path:** . (text field)
- Execute @Dest Env?:** Yes No
- Execute as 'apps?':** Yes No

At the bottom of the dialog are four buttons: "Clear", "OK", "Add", and "Cancel". A status bar at the very bottom of the window reads "SQL Loader File' parameters loaded."

Table 3-9. SQL Loader File object type field descriptions (page 1 of 2)

Field Name (*Required)	Description
Control file (.CTL)	SQL Loader control file name.
*Sub-Path	Subdirectory (relative to the base directory) for the environment where the control file is located.
Execute @Dest Env	If set to Yes , SQL Loader automatically loads data at the destination instance.
Execute as 'apps'	If set to Yes , and if HP Deployment Management Extension for Oracle E-Business Suite is installed, SQL Loader uses the username and password for the APPS (main database) user to load the data at the destination instance. If HP Deployment Management Extension for Oracle E-Business Suite is not installed, this field can still be used to differentiate between a central database login and an application-specific one.
Log file (Hidden by default)	Name of the log file generated by SQL Loader.
Bad file (Hidden by default)	Name of the file containing bad rows.
Data file (Hidden by default)	Data file name, if an external data file exists.
# of logical recs to skip (Hidden by default)	Number of logical records to skip.
Use Direct path (Hidden by default)	Option to use direct paths.
Max. num of discards (Hidden by default)	Maximum number of records that can be discarded before the load stops.
Discard file (Hidden by default)	Name of the file where discarded records are to be placed.
Extents file Hidden by default)	File from which to allocate extents.

Table 3-9. SQL Loader File object type field descriptions (page 2 of 2)

Field Name (*Required)	Description
Max # of errors (Hidden by default)	Maximum number of errors allowed before the load stops.
# of logical recs to load (Hidden by default)	Number of logical records to load.
Parameter file (Hidden by default)	Name of the file that contains the parameter specifications.
Run in silent mode (Hidden by default)	Option to suppress messages (header, feedback, errors, discards) during the run.

SQL Loader80 File Object Type

The SQL Loader80 File object type migrates the Oracle SQL Loader control file from one Oracle instance to another. The object type then optionally runs SQL Loader to load the data into an Oracle database at the destination instance.

Figure 3-9 shows the default screen when adding a package line that uses the SQL Loader80 File object type. *Table 3-10* provides field descriptions for the object type.



The SQL Loader80 File object type does not, by default, use any of the fields in *Table 3-10* that are not required. These fields are provided to allow users to customize and add to the SQL Loader call within the command steps.

Figure 3-9. SQL Loader80 File object type sample data

Table 3-10. SQL Loader80 File object type field descriptions (page 1 of 2)

Field Name (*Required)	Description
Control file (.CTL)	SQL Loader control file name.
*Sub-Path	Subdirectory (relative to the base directory) for the environment where the control file is located.
Execute @Dest Env	If set to Yes , SQL Loader automatically loads data at the destination instance.
Execute as 'apps'	If set to Yes , and if HP Deployment Management Extension for Oracle E-Business Suite is installed, SQL Loader uses the username and password for the APPS (main database) user to load the data at the destination instance. If HP Deployment Management Extension for Oracle E-Business Suite is not installed, this field can still be used to differentiate between a central database login and an application-specific one.

Table 3-10. SQL Loader80 File object type field descriptions (page 2 of 2)

Field Name (*Required)	Description
Log file (Hidden by default)	Name of the log file generated by SQL Loader.
Bad file (Hidden by default)	Name of the file containing bad rows.
Data file (Hidden by default)	Data file name, if an external data file exists.
# of logical recs to skip (Hidden by default)	Number of logical records to skip.
Use Direct path (Hidden by default)	Option to use direct paths.
Max. num of discards (Hidden by default)	Maximum number of records that can be discarded before the load stops.
Discard file (Hidden by default)	Name of the file where discarded records are to be placed.
Extents file (Hidden by default)	File from which to allocate extents.
Max # of errors (Hidden by default)	Maximum number of errors allowed before the load stops.
# of logical recs to load (Hidden by default)	Number of logical records to load.
Parameter file (Hidden by default)	Name of the file that contains the parameter specifications.
Run in silent mode (Hidden by default)	Option to suppress messages (header, feedback, errors, discards) during the run.

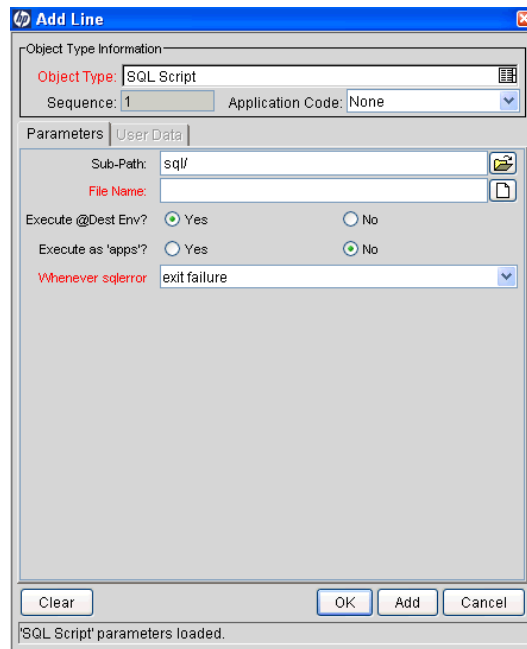
SQL Script Object Type

The SQL Script object type moves one or more SQL or PL/SQL scripts from one instance to another, and optionally executes the scripts at the destination. If you are running Oracle Applications with the APPS account as the default username, then the SQL Script object type has the ability to execute the script as APPS after the script has been migrated to the destination environment.

You can modify this object type to include steps to interact with your version control system.

Figure 3-10 shows the default screen when adding a package line that uses the SQL Script object type. *Table 3-11* provides field descriptions for the object type.

Figure 3-10. SQL Script object type sample data



The screenshot shows a dialog box titled "Add Line" with a blue header bar. The main area is divided into two sections: "Object Type Information" and "Parameters".

Object Type Information:

- Object Type: SQL Script (text field)
- Sequence: 1 (text field)
- Application Code: None (dropdown menu)

Parameters:

- Sub-Path: sql/ (text field with folder icon)
- File Name: (text field with folder icon)
- Execute @Dest Env?: Yes No
- Execute as 'apps?': Yes No
- Whenever sqlerror: exit failure (dropdown menu)

At the bottom, there are buttons for "Clear", "OK", "Add", and "Cancel". A status bar at the very bottom reads: "SQL Script parameters loaded."

Table 3-11. SQL Script object type field descriptions

Field Name (*Required)	Description
Sub-Path	Subpath (relative to base path) to the SQL script.
*File Name	Name of the SQL script to be migrated.
Execute @Dest Env	Option to execute the SQL script.
Execute as 'apps'	If set to Yes , and if HP Deployment Management Extension for Oracle E-Business Suite is installed, SQL Loader uses the username and password for the APPS (main database) user to load the data at the destination instance. If HP Deployment Management Extension for Oracle E-Business Suite is not installed, this field can still be used to differentiate between a central database login and an application-specific one.
*Whenever sqlerror	Whether the object type should exit failure or continue if an error is encountered while the script is executing.

Index

- D**
 - ddlDriver.<#####>.log, [16](#)
- F**
 - fmb file extension, [22](#), [23](#), [25](#)
 - fmt file extension, [22](#), [23](#), [25](#)
 - fmX file extension, [22](#), [23](#), [25](#)
 - Forms 10G object type, [25](#)
 - Forms 4.5 object type, [22](#)
 - Forms 6.0 object type, [23](#)
- I**
 - installation, [11](#)
 - collecting information for, [13](#)
 - logs generated during installation, [16](#)
 - post-installation procedures, [18](#)
 - preparation, [13](#)
 - procedure, [15](#)
 - summary report, [16](#)
 - system requirements, [11](#)
 - ITG_DEPLOYMENT_HOME environment variable, [14](#)
- J**
 - jarxvf.<#####>.log, [16](#)
- K**
 - kDeploy.sh script, [15](#), [17](#)
- L**
 - logs generated during installation, [16](#)
 - ddlDriver.<#####>.log, [16](#)
 - jarxvf.<#####>.log, [16](#)
 - packageDriver.<#####>.log, [17](#)
 - postXMLDriver.<#####>.log, [17](#)
 - preXMLDriver.<#####>.log, [17](#)
- O**
 - object types
 - Forms 10G, [25](#)
 - Forms 4.5, [22](#)
 - Forms 6.0, [23](#)
 - Reports 10G, [30](#)
 - Reports 2.0, [26](#)
 - Reports 2.5, [28](#)
 - Reports 6.0, [29](#)
 - SQL Loader File, [32](#)
 - SQL Loader80 File, [34](#)
 - SQL Script, [37](#)
 - Oracle Forms, [7](#), [8](#)
 - Oracle Reports, [7](#), [8](#)
 - Oracle SQL Loader, [32](#), [34](#)
- P**
 - packageDriver.<#####>.log, [17](#)
 - PL/SQL, [7](#), [8](#)
 - PL/SQL scripts, moving, [37](#)
 - postXMLDriver.<#####>.log, [17](#)

PPM_Home directory, **14**
prerequisite knowledge and experience, **10**
preXMLDriver.<#####>.log, **17**

R

rdf file extension, **26, 28, 29, 30**
reference entities, **12**
Reports 10G object type, **30**
Reports 2.0 object type, **26**
Reports 2.5 object type, **28**
Reports 6.0 object type, **29**
requirements, system, **11**
rex file extension, **26, 28, 29, 30**

S

scripts
 kDeploy.sh, **15, 17**
 setServerMode.sh, **14, 18**
setServerMode.sh script, **14, 18**
SQL Loader File object type, **32**
SQL Loader80 File object type, **34**
SQL Loader, Oracle, **32, 34**
SQL Script object type, **37**
SQL scripts, moving, **37**
SQL*Plus, **7, 8**
system requirements, **11**

U

upgrade, **11**
 impacts and guidelines, general, **12**
 impacts, version 9.10, **13**
 preparation, **13**

V

version 9.10, what's new and changed, **8**