Oracle® Retail Predictive Application Server

Installation Guide

Release 14.1.2

E70811-01

January 2016

ORACLE®

Oracle® Retail Predictive Application Server Installation Guide, 14.1.2

E70811-01

Primary Author: Melissa Artley

reconfigurations, reassembly or reverse assembly, re-engineering or reverse engineering and recompilations or reverse compilations of the VAR Applications or any derivatives of the VAR Applications. You acknowledge that it shall be a breach of the agreement to utilize the relationship, and/or confidential information of the VAR Applications for purposes of competitive discovery.

The VAR Applications contain trade secrets of Oracle and Oracle's licensors and Customer shall not attempt, cause, or permit the alteration, decompilation, reverse engineering, disassembly or other reduction of the VAR Applications to a human perceivable form. Oracle reserves the right to replace, with functional equivalent software, any of the VAR Applications in future releases of the applicable program.

Contents

3 Installing on UNIX and Linux Environments

4 Installing on a Windows Environment

5 Installing the RPAS Fusion Client

6 Installing and Configuring the RPAS Classic Client

7 RPAS Classic Client Web Deployment

8 Installing and Building the RPAS HSA Environment

D Appendix: Oracle Single Sign-On (SSO)

E Appendix: Installing Solution Plug-ins

F Appendix: Installation Order

List of Tables

List of Figures

Send Us Your Comments

Oracle Retail Predictive Application Server Installation Guide, Release 14.1.2.

Oracle welcomes customers' comments and suggestions on the quality and usefulness of this document.

Your feedback is important, and helps us to best meet your needs as a user of our products. For example:

- Are the implementation steps correct and complete?
- Did you understand the context of the procedures?
- Did you find any errors in the information?
- Does the structure of the information help you with your tasks?
- Do you need different information or graphics? If so, where, and in what format?
- Are the examples correct? Do you need more examples?

If you find any errors or have any other suggestions for improvement, then please tell us your name, the name of the company who has licensed our products, the title and part number of the documentation and the chapter, section, and page number (if available).

> **Note:** Before sending us your comments, you might like to check that you have the latest version of the document and if any concerns are already addressed. To do this, access the Online Documentation available on the Oracle Technology Network Web site. It contains the most current Documentation Library plus all documents revised or released recently.

Send your comments to us using the electronic mail address: retail-doc_us@oracle.com.

Please give your name, address, electronic mail address, and telephone number (optional).

If you need assistance with Oracle software, then please contact your support representative or Oracle Support Services.

If you require training or instruction in using Oracle software, then please contact your Oracle local office and inquire about our Oracle University offerings. A list of Oracle offices is available on our Web site at http://www.oracle.com.

Preface

Oracle Retail Installation Guides contain the requirements and procedures that are necessary for the retailer to install Oracle Retail products.

Audience

This Installation Guide is for the following audiences:

- Database administrators (DBA)
- System analysts and designers
- Integrators and implementation staff

Documentation Accessibility

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

Access to Oracle Support

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

Related Documentation

For more information, refer to the following documents in the Oracle Retail Predictive Application Server Release 14.1.2 documentation set:

- *Oracle Retail Predictive Application Server Administration Guide for the Classic Client*
- *Oracle Retail Predictive Application Server Administration Guide for the Fusion Client*
- *Oracle Retail Predictive Application Server Installation Guide*
- *Oracle Retail Predictive Application Server Release Notes*

Customer Support

To contact Oracle Customer Support, access My Oracle Support at the following URL:

https://support.oracle.com

When contacting Customer Support, please provide the following:

- Product version and program/module name
- Functional and technical description of the problem (include business impact)
- Detailed step-by-step instructions to re-create
- Exact error message received
- Screen shots of each step you take

Review Patch Documentation

When you install the application for the first time, you install either a base release (for example, 14.1) or a later patch release (for example, 14.1.2). If you are installing the base release, additional patch, and bundled hot fix releases, read the documentation for all releases that have occurred since the base release before you begin installation. Documentation for patch and bundled hot fix releases can contain critical information related to the base release, as well as information about code changes since the base release.

Improved Process for Oracle Retail Documentation Corrections

To more quickly address critical corrections to Oracle Retail documentation content, Oracle Retail documentation may be republished whenever a critical correction is needed. For critical corrections, the republication of an Oracle Retail document may at times not be attached to a numbered software release; instead, the Oracle Retail document will simply be replaced on the Oracle Technology Network Web site, or, in the case of Data Models, to the applicable My Oracle Support Documentation container where they reside.

This process will prevent delays in making critical corrections available to customers. For the customer, it means that before you begin installation, you must verify that you have the most recent version of the Oracle Retail documentation set. Oracle Retail documentation is available on the Oracle Technology Network at the following URL:

http://www.oracle.com/technetwork/documentation/oracle-retail-100266.html

An updated version of the applicable Oracle Retail document is indicated by Oracle part number, as well as print date (month and year). An updated version uses the same part number, with a higher-numbered suffix. For example, part number E123456-02 is an updated version of a document with part number E123456-01.

If a more recent version of a document is available, that version supersedes all previous versions.

Oracle Retail Documentation on the Oracle Technology Network

Oracle Retail product documentation is available on the following web site:

http://www.oracle.com/technetwork/documentation/oracle-retail-100266.html

(Data Model documents are not available through Oracle Technology Network. You can obtain them through My Oracle Support.)

Conventions

The following text conventions are used in this document:

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

1

Introduction

Welcome to the *Oracle Retail Predictive Application Server Installation Guide*. This chapter outlines the contents of this guide, discusses the updated components with respect to the previous version, lists hardware and software requirements, and defines commonly used notations and terms.

About This Document

This document contains information for full and patch installations:

- Full Installation. Refer to this section if you are performing a full installation of RPAS 14.1.2.
- Patch Installation. Refer to this section if you are performing a patch installation of RPAS 14.1.2.

> **Note:** Only RPAS base releases have installers. Patch releases and hotfixes do not have installers. If you are installing a base release, refer to Full Installation for installer instructions. If you are installing a patch, refer to Patch Installation for patch instructions.

Read this entire guide before beginning the installation process to ensure you understand the installation process and have all the necessary documentation, hardware, and software available.

Hardware and Software Requirements

Table 1–1 describe the hardware and software requirements for the RPAS Server, RPAS Configuration Tools, and ODBC and JDBC servers and clients.

RPAS Server and Components

For information on installing the RPAS Server, refer to either Chapter 3, "Installing on UNIX and Linux Environments" or Chapter 4, "Installing on a Windows Environment".

Table 1–1 RPAS Server and Components Hardware and Software Requirements

Component	Details
Supported Operating Systems for the RPAS Server, Compilers, ODBC Server, and ODBC Client	AIX 7.1 TL3 SP1 IBM XLC 12.1 (64 bit)
	Oracle Enterprise Linux 6.5. Red Hat Enterprise Linux 6.5 Linux Compiler: gcc 4.4.6 (ships standard with Oracle Linux 6.5)
	Oracle Solaris 11.1 (SPARC) Solaris Studio 12.3
	HP-UX Itanium 11.31 HP-UX Compiler: aCC A.06.20
	Note: The RPAS ODBC Client is supported on the previously listed operating systems, as well as 32-bit Windows or 64-bit Windows. **Note:** The C++ compilers are only needed for building custom extension libraries, and not for using base RPAS Platform and Application functionality. **Note:** The ksh shell is required on all operating systems for the RPAS server.
Supported Operating Systems for RPAS Configuration Tools	Microsoft Windows 7 Microsoft Windows 8.1
	Note: The server aspect of the RPAS Configuration Tools (rpasInstall, which is used to install and patch domains) is supported on the same operating systems as the previously listed RPAS Server.
Required Software	Java 7 is required to support the RPAS Configuration Tools, server machines, and JDBC Client. There are specific Java versions supported for each of the following operating systems: ■ AIX 7.1 TL3 SP1: IBM Java 1.7.0.R1.SP9.FP10(64bit) ■ Oracle Linux 6.5: Oracle Java 1.7u91 (64-bit) ■ Red Hat Enterprise Linux 6.5: Oracle Java 1.7u91 (64-bit) ■ Oracle Solaris 11.1:Solaris Studio 12.3: Oracle Java 1.7u91 (64-bit) ■ HP-UX 11.31 Itanium: HP Java 1.7.0_14ia (64bit) ■ Microsoft Windows 7 ■ Microsoft Windows 8.1 **Note:** A full JDK installation is required only if building Java-based custom extensions for RPAS. For running RPAS, only the JRE is required.

RPAS Classic Client

Table 1–2 describe the hardware and software requirements for the RPAS Classic Client. For information on installing the RPAS Classic Client, refer to Chapter 6, "Installing and Configuring the RPAS Classic Client". For information about Single Sign-On, refer to "Appendix: Oracle Single Sign-On (SSO)".

Table 1–2 RPAS Classic Client Hardware and Software Requirements

Component	Details
Supported Operating Systems	Microsoft Windows 7 Professional with Microsoft Office 2010 or 2013
Client System Requirements	**All components are required:** ■ 800x600 or higher display resolution ■ 1GHz or higher processor ■ 512 MB or higher memory ■ Intranet network connectivity with at least 10Mbps data rate
RPAS Classic Client Web Deployment and Single Sign-On (SSO) Requirements	**Note:** Classic Client Web Deployment, along with SSO, is optional.
	Server options (only one of these options is required): ■ Oracle WebLogic Server 12c (Release 12.1.3) with Oracle Application Development Framework (12.1.3 (12.1.3 with patch 19473060) and Java 1.7.0.R1.SP9.FP10. If Oracle SSO is used, the Web Tier Utilities 12.1.3 must be installed. ■ Apache Tomcat 6.0 with Java 1.7.0.R1.SP9.FP10
	Web browser requirements: ■ Microsoft Internet Explorer 11 ■ Google Chrome 40 ■ Oracle JVM plug-in of Java version 1.7.0.R1.SP9.FP10 ■ Mozilla Firefox 38 ESR
	Single Sign-On (SSO) WebLogic requires both of these components: ■ Oracle Web Tier Utilities (12.1.3) ■ Oracle Identity Management 11g (11.1.1.7)
	RPAS Classic Client Web Deployment is certified for SSO using Oracle Access Manager 11g (OAM). This is available in the Oracle Identity and Access Management 11g package (11.1.2.2).
	OAM works with a wide variety of LDAP-based identity stores. RPAS Classic Client Web Deployment has been verified to work with Oracle Internet Directory 11g (OID 11.1.1.7).
	OAM-based SSO needs a web server. RPAS Classic Client Web Deployment has been verified to work with Oracle HTTP Server (OHS) 12.1.3.
	A web server plug-in (OAM Webgate) communicates with an OAM instance and to create the SSO cookie. OAM Webgate 11.1.2.2 is required.
	In summary, the following Oracle Fusion Middleware packages are utilized in the solution: ■ Oracle Web Tier Utilities (12.1.3) ■ Oracle Identity Management 11g (11.1.1.7) ■ Oracle Identity and Access Management 11g (11.1.2.2) ■ Oracle Access Manager Webgate 11g (11.1.2.2)

RPAS Fusion Client

Table 1–3 describe the hardware and software requirements for the RPAS Fusion Client. For information on installing the RPAS Fusion Client, refer to Chapter 5, "Installing the RPAS Fusion Client".

Table 1–3 RPAS Fusion Client Hardware and Software Requirements

Component	Details
Supported Operating Systems for the Fusion Client	Microsoft Windows 7 Professional with Microsoft Office 2010 or 2013
Supported Web Browsers	**Note:** All browsers must have Adobe Flash Player 9.0 (or later).
	Microsoft Windows 7 ■ Microsoft Internet Explorer 11 ■ Google Chrome 40 ■ Mozilla Firefox 38 ESR
Oracle Database	Oracle 12c Database **Note:** The WebLogic Server 12c (Release 12.1.3) and ADF use Oracle Platform Security Services (OPSS); use by the Oracle 12c Database is not required of the Fusion Client application.
Application Server Requirements	All components required: ■ Oracle WebLogic Server 12c (Release 12.1.3) ■ Oracle Application Development Runtime 12c (Release 12.1.3); apply both ADF Patches 18886249, 19473060, and 21832568 (downloaded from My Oracle Support)
Supported Operating Systems for the Application Server	**Note:** The operating systems you choose for the RPAS server and the application server for the RPAS Fusion Client do not need to be the same.
	AIX 7.1 TL3 SP1
	Oracle Linux 6.5 (64 bit) Red Hat Enterprise Linux 6.5
	Oracle Solaris 11.1 (SPARC) Solaris Studio 12.3)
	HP-UX Itanium 11.31
Required Software	Java 7 is required to support the Fusion Client on an application server. There are specific JDK versions supported for each of the following operating systems: ■ IBM AIX 7.1 TL3 SP1 IBM Java 1.7.0.R1.SP9.FP10 (64-bit) ■ Oracle Solaris 11.1:Solaris Studio 12.3: Oracle Java 1.7u91 (64-bit) ■ Oracle Linux 6.5: Oracle Java 1.7u91 (64-bit) ■ Red Hat Enterprise Linux 6.5: Oracle Java 1.7u91 (64-bit) ■ HP-UX 11.31 Itanium: HP Java 1.7.0_14ia (64-bit) **Note:** A full JDK installation is required only if building Java-based custom extensions for RPAS. For running RPAS, only the JRE is required.

Table 1–3 (Cont.) RPAS Fusion Client Hardware and Software Requirements

Component	Details
Supported Oracle Software for Single Sign-On (SSO)	**Note:** SSO is optional.
	RPAS Fusion Client is certified for SSO using Oracle Access Manager 11g (OAM). This is available in the Oracle Identity and Access Management 11g package (11.1.2.2).
	OAM works with a wide variety of LDAP-based identity stores. RPAS Fusion Client in SSO deployment has been verified to work with Oracle Internet Directory 11g (OID 11.1.1.7).
	OAM-based SSO needs a web server. RPAS Fusion Client has been verified to work with Oracle HTTP Server (OHS) 12.1.3.
	A web server plug-in (OAM Webgate) communicates with an OAM instance and to create the SSO cookie. OAM Webgate 11.1.2.2 is required.
	In summary, the following Oracle Fusion Middleware packages are utilized in the solution: ■ Oracle Web Tier Utilities (12.1.3) ■ Oracle Identity Management 11g (11.1.1.7) ■ Oracle Identity and Access Management 11g (11.1.2.2) ■ Oracle Access Manager Webgate 11g (11.1.2.2)

Hardware and Software Requirement Notes

The following notes pertain to the RPAS hardware and software requirements:

- Once the RPAS server and client are installed, you must build and set up an RPAS domain or install an RPAS solution. For more information on setting up an RPAS domain, refer to either the Classic Client or Fusion Client version of the *Oracle Retail Predictive Application Server Administration Guide*. For more information on installing an RPAS solution, refer to the relevant installation guide included with the RPAS solution package.
- If you are installing only the Oracle RPAS Fusion Client without any additional applications supported by the Oracle Application Development Runtime, you still need to install the Oracle Database repository schema specified by the Oracle Application Development Runtime installation instructions.
- If you are installing the RPAS Server on Windows, you must install Cygwin to emulate UNIX commands (required for starting the RPAS Server on Windows). You can find more information about downloading this product at `http://www.cygwin.com`. If running the RPAS Server on Windows, use Windows 7. Perl is an interpreted language that is included on all supported UNIX platforms (included with Cygwin for Windows). Perl is used by the patch sets, which are used to install an RPAS patch.
- An application for unzipping (`.zip`) components on UNIX must be installed and used for extracting the RPAS Configuration Tools. Unzip is an open source software package that can be used for this process.

Table 1–4 indicates which software components are needed for each task. The reference to Windows refers to Windows 7.

Table 1–4 Software Requirements Notes

Task	Typical User	Platforms	RPAS Server	RPAS Clients	Configuration Tools	Java
Log in to an existing (built) RPAS domain for the primary purpose of building workbooks	End User	Windows	No	Yes	No	No
Use the Configuration Tools to create or modify solutions.	Solution/ Product Administrator	Windows	Yes	No	Yes	Yes
Use the Configuration Tools to build configured solutions.	Solution/ Product Administrator	Windows UNIX Linux	Yes	No	Yes	Yes

Java can be acquired from www.java.com for Oracle Solaris and Microsoft Windows or from the respective vendor's Web site for IBM and HP.

Environment variables are automatically set when using the Oracle Retail Installer to install the RPAS components on a UNIX environment.

Monitor Disk Space

Array corruption can occur if arrays are truncated due to lack of disk space. To prevent such array corruptions, the following setup is recommended:

Setup an alert to monitor free space available on the disk. If the available disk space falls below a certain threshold, an alert needs to be triggered. These alerts can be setup using simple shell scripts.

Supported Oracle Retail Products

For information on the version of the RPAS platform that an RPAS application uses, refer to that application's documentation.

Requesting Infrastructure Software

If you are unable to find the necessary version of the required Oracle infrastructure software (database server, application server, WebLogic, and so on.) on the Oracle Software Delivery Cloud, you should file a non-technical 'Contact Us' Service Request (SR) and request access to the media. For instructions on filing a non-technical SR, see My Oracle Support Note 1071023.1 – *Requesting Physical Shipment or Download URL for Software Media*.

Terms

The following table lists terms that are used in this guide.

Term	Definition
RPAS	The Oracle Retail Predictive Application Server provides the foundation for Oracle Retail solutions such as Oracle Retail Demand Forecasting (RDF), Merchandise Financial Planning (MFP), and Advanced Inventory Planning (AIP). RPAS does not include any business logic, but it enables the solutions to store, manipulate and retrieve data. It provides the solutions with a standard interface based on wizards, templates, workbooks, and batch processes.

Term	Definition
RPAS solution	The software that uses RPAS. RPAS solutions are added on to RPAS domains as separate modules. All the business logic is encapsulated in the solution. An RPAS domain can support multiple solutions.
RPAS domain	The collection of server-side directories and files containing data and procedures that comprise the RPAS solution. For additional information, refer to the *Oracle Retail Predictive Application Server Configuration Tools User Guide* and the Classic Client and Fusion Client versions of the *Oracle Retail Predictive Application Server Administration Guide*.
RPAS Classic Client	The Windows-based client interface for end users and system administrators of an RPAS domain. An administrator may perform maintenance work in a domain using the RPAS Classic Client, server-side RPAS utilities.
RPAS Fusion Client	The Web-based Rich Client for end users and system administrators of an RPAS domain. An administrator may perform maintenance work in a domain using the RPAS Fusion Client in a browser. This Retail Predictive Application Server (RPAS) platform is developed using the Oracle Application Development Framework (ADF).
RPAS Configuration Tools	The tools used to configure an RPAS solution. For more information, refer to the *Oracle Retail Predictive Application Server Configuration Tools User Guide*.

Part I

Full Installation

Part I of this guide details the steps needed to perform a full installation of RPAS.

Part I contains the following chapters:

Chapter 2, "Getting Started"

Chapter 3, "Installing on UNIX and Linux Environments"

Chapter 4, "Installing on a Windows Environment"

Chapter 5, "Installing the RPAS Fusion Client"

Chapter 6, "Installing and Configuring the RPAS Classic Client"

Chapter 7, "RPAS Classic Client Web Deployment"

Chapter 8, "Installing and Building the RPAS HSA Environment"

For information about a patch installation, refer to Patch Installation.

2

Getting Started

This chapter provides:

- An overview of the RPAS platform
- Typical installation scenarios
- An overview of the installation contents

RPAS Platform Overview

RPAS 14.1.2 is comprised of many components. In addition, there are solutions that have been developed using the RPAS 14.1 foundation. These solutions must be installed separately. Examples of these solutions include Oracle Retail Merchandise Financial Planning (MFP) and Oracle Retail Advanced Inventory Planning (AIP).

The components of the RPAS software include the following:

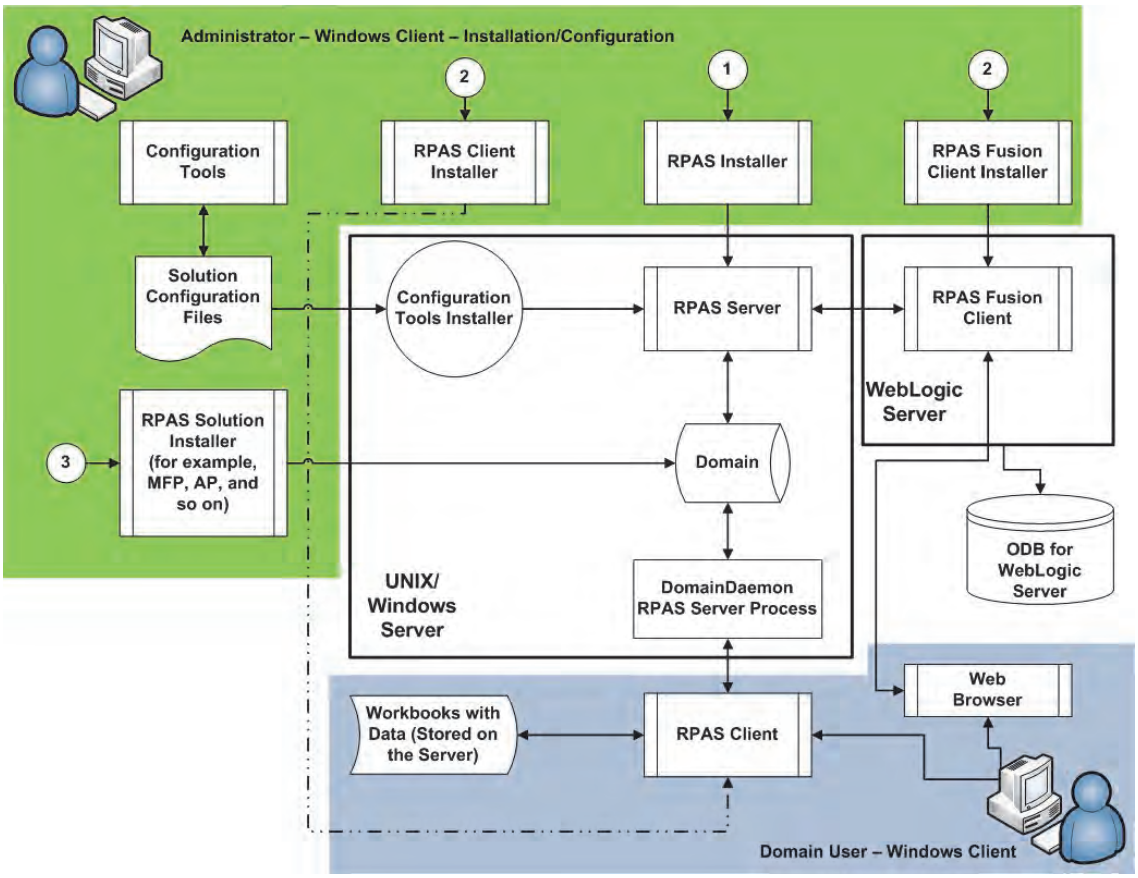
- RPAS Server and related utilities
- RPAS Classic Client
- RPAS Fusion Client
- RPAS Configuration Tools
- Sample configurations (Curve and Grade)
- Documentation
- Supported Translations

> **Note:** The use of the RPAS Fusion Client and RPAS Classic Client simultaneously in the same environment is not supported in a production environment.

Installation Process Flow

A typical RPAS Server-based installation is illustrated in Figure 2–1. For instructions on installing on a Windows machine, refer to Chapter 4, "Installing on a Windows Environment".

Figure 2–1 Installation Process Flow



RPAS Environment

Figure 2–1 displays a typical RPAS Server-based installation and provides the following information:

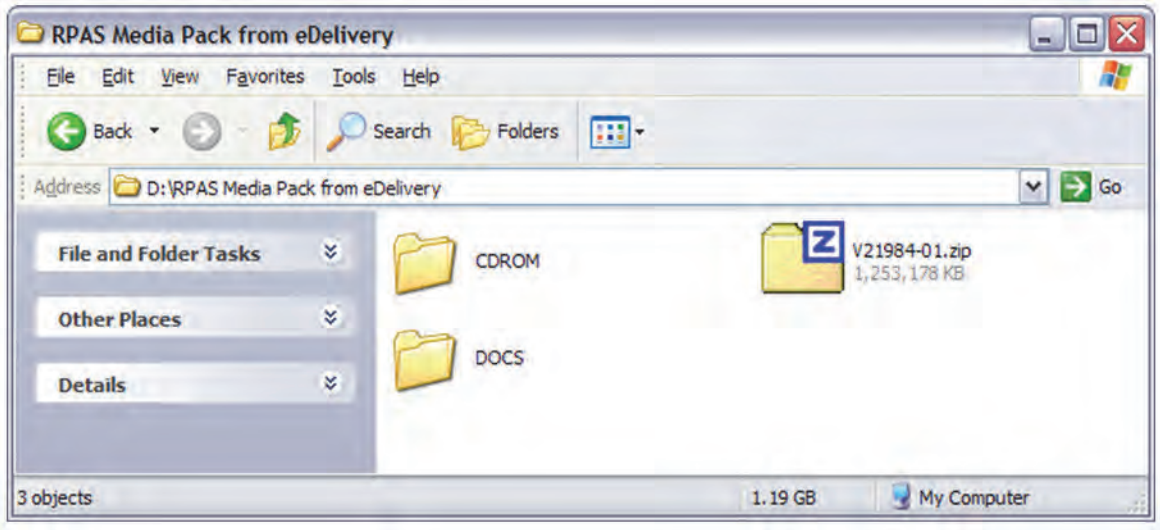
- RPAS and the Configuration Tools may also be installed on a single Windows stand-alone machine.
- Numbers represent the order in which you must install the RPAS components.
- The RPAS Classic Client Installer and RPAS Fusion Client Installer processes have the same number. This indicates that you can choose to install the RPAS Classic Client on the domain user systems or the RPAS Fusion Client on an Oracle WebLogic Server instance.
- Before you install an RPAS solution, you must have the RPAS Server and RPAS Clients installed. The RPAS solution installers include the taskflow configuration and online help files that require the RPAS Fusion Client to be already installed.
- A domain user can log on to an RPAS domain/solution using the RPAS Classic Client or by logging on to the RPAS Fusion Client through a Web browser.

Downloading and Extracting the RPAS Media Pack

The following procedure provides information about extracting the RPAS Media Pack and its contents.

1. Create a directory to store the RPAS Media Pack and download the media pack to this location.
2. Extract the media pack to this location. Once extracted, two folders appear, **CDROM** and **DOCS**.

Figure 2–2 Example of CDROM and DOCS Folder Created by Extracting Media Pack ZIP File



The **CDROM** folder contains the ZIP files listed in Table 2–1.

Table 2–1 ZIP File Contents

File	Contents
FusionClient.zip	The RPAS Fusion Client installer.
RPAS-14.1.2-unix.zip	All the RPAS components to be installed on your UNIX server.
RPAS-14.1.2-windows.zip	The RPAS components that can be installed on a Windows environment. Refer to the chapters, "Installing on UNIX and Linux Environments" and "Installing on a Windows Environment" for information on performing installations.

The **DOCS** folder has folders within it that contain the RPAS documentation in PDF format.

RDF Packaging

The scripts for Oracle Retail Demand Forecasting (RDF) are packaged within the RPAS/RDF server installation. For more information, refer to the RDF documentation.

The Curve and Grade documentation sets which were previously included in the **DOCS** folder are now combined with the RDF documentation. This documentation is available within the RDF package and on the following Web site:

`http://www.oracle.com/technology/documentation/oracle_retail.html`

3

Installing on UNIX and Linux Environments

The installation of the server-side RPAS components on UNIX or Linux operating systems is accomplished by using a Java-based installation program that is included with the installation package.

This program automates the following:

- Installation of the RPAS Server
- Installation of Configuration Tools on the server
- Creation of sample domains
- Definition of DomainDaemon port

Preparation

The RPAS components included in this installation process are available inside the media pack of the solution downloaded from Oracle's E-Delivery Web site, `http://edelivery.oracle.com/`.

There are two RPAS archives inside the media pack: one RPAS archive for UNIX and Linux and one for Windows.

Java Environment

Ensure that Java Development Kit (JDK) has been installed on the machine where RPAS will run and that the JAVA_HOME environment variable is properly set.

> **Note:** Avoid enabling AutoUpdate when installing Java because it may update the Java version without prompting.

The following commands are examples based on common Java installation structures. These commands are automatically setup by the `retaillogin.ksh` script. After this script has run, compare your environment variables to the appropriate subsequent examples to ensure that it is correct. To ensure the path is correct, use the UNIX find command to locate libjvm.so (find `$JAVA_HOME -name libjvm.so`) and adjust the following paths as appropriate.

Linux

Example 3–1 shows a sample of common Java installation structures for Linux.

Example 3–1 Linux

```
export LD_LIBRARY_PATH="$JAVA_HOME/lib:$JAVA_HOME/jre/lib/amd64/server:
$LD_LIBRARY_PATH"
```

AIX

Example 3–2 shows a sample of common Java installation structures for AIX.

Example 3–2 AIX

```
export LIBPATH="$JAVA_HOME/lib/ppc64:$JAVA_HOME/bin/classic:$LIBPATH"
export PATH="$JAVA_HOME/bin:$PATH"
```

Note: Linux and AIX are compatible with only a 64-bit version of Java.

Solaris

Example 3–3 shows a sample of common Java installation structures for Solaris.

Example 3–3 Solaris

```
export LD_LIBRARY_PATH="$JAVA_HOME/lib:$JAVA_HOME/jre/lib/sparcv9/server:$JAVA_
HOME/jre/lib/sparc: $LD_LIBRARY_PATH"
export RIDE_OPTIONS=-d64
```

Note: The PATH variable on Solaris operating systems must have /usr/xpg4/bin in the path, and there can be no other occurrence of /bin or /usr/bin to its left. Otherwise, the scripts use the incorrect versions of grep, sed, awk, tail, and so on.

HP-UX Itanium

Example 3–4 shows a sample of common Java installation structures for HP-UX Itanium.

Example 3–4 HP-UX Itanium

```
export SHLIB_PATH="$JAVA_HOME/lib:$JAVA_HOME/jre/lib/IA64W/server:$JAVA_
HOME/jre/lib/IA64W/:$SHLIB_PATH"
export PATH="$JAVA_HOME/bin:$PATH"
export RIDE_OPTIONS=-d64
```

Note: Solaris and HP Itanium do not release separate 32-bit and 64-bit versions of Java. Therefore, you must set the 64-bit Configuration Tools environment variable for Java as shown:

```
export RIDE_OPTIONS=-d64
```

RPAS_JAVA_CLASSPATH

The RPAS_JAVA_CLASSPATH variable is used by RPAS applications that use the RPAS Java Special Expression functionality. This variable tells RPAS where to find the Java Special Expression implementations. It should contain a list of the locations of any

Java code that is needed to run a Java Special Expression. The list must also include the `oracleRpasUtils.jar` located in the **lib** subdirectory of **RPAS_HOME**. For information about exporting RPAS_JAVA_CLASSPATH through Cygwin, refer to Cygwin Path Settings.

Example 3–5 RPAS_JAVA_CLASSPATH

In the following example, the absolute path to `oracleRpasUtils.jar` file name is listed as well as an example of a Java Special Expression implementation where customJavaExpression.jar contains the example expression.

```
export RPAS_JAVA_
CLASSPATH="/Oracle/RPAS/lib/oracleRpasUtils.jar;/Oracle/RPAS/applib/customJavaExpr
ession.jar"
```

> **Note:** Certain RPAS applications may have additional uses for the RPAS_JAVA_CLASSPATH environment variable. To determine if you need to set this variable in a different way, refer to that application's Installation Guide.

Ride Options

The RIDE_OPTIONS environmental variable has been defined to allow users to pass information into the rpasInstall process. Unlike the regular arguments passed on the command line to rpasInstall (such as `-fullinstall` and `-updatestyles`), arguments defined in RIDE_OPTIONS are passed to every rpasInstall instance that runs in the environment.

There are the three supported properties for use with RIDE_OPTIONS:

- `Xmx` — used for Java
- `HP 64-bit mode Java (-d64)` — used for HP Itanium and Solaris
- `Drpas.maxProcesses` — used for RPAS

For Java

`Xmx` - By default, the Java Virtual Machine requests on the order of 268 MB of RAM from the OS to allocate for its heap. Large domains that are built from complex configurations can potentially exhaust this limited amount of memory. This is even more of an issue in patch installations than in builds since a patch installation requires two different versions of a configuration to be held in memory simultaneously.

By using the `-Xmx` option, you can instruct the Java Virtual Machine to request more memory from the OS to prevent situations when all allocated memory is exhausted. The syntax of the property is:

`-Xmx###m`, where ### is the amount, in megabytes, of memory the JVM is to request. Common values for this argument are `-Xmx512m` or `-Xmx1024m`.

For HP Itanium and Solaris

`HP 64-bit mode Java (-d64)` - The HP and Solaris distribution of Java does not consist of separate executables for 32-bit and 64-bit operating systems. Instead, there is a single distribution of Java that can run in either 32-bit or 64-bit mode. By default, the HP Java runs in 32-bit mode. Because RPAS is built as a 64-bit executable on the Itanium OS, the RPAS libraries are unable to link with Java if it is running in 32-bit mode.

By adding the -d64 property to RIDE_OPTIONS, the HP Java distribution is 64-bit mode enabled and the RPAS libraries link successfully.

It is often the case that users may want to use other properties in conjunction with RIDE_OPTIONS. When this is the case, all desired properties should be included within the environmental variable definition separated by white space with the entire definition enclosed in double quotes. An example of this is shown as:

```
export RIDE_OPTIONS="-d64 -Xmx1024m -Drpas.maxProcesses=8"
```

For RPAS

Drpas.maxProcesses - Several RPAS server utilities are designed to take advantage of multi-processor hardware to improve their performance. These utilities attempt to perform operations in parallel, each process running on a distinct processor.

The -Drpas.maxProcesses argument is used to instruct RPAS how many processors it should attempt to run in parallel when running one of the server utilities that has multi-processor support when that utility is run as a part of the rpasInstall process.

Note that the -Drpas.maxProcesses argument only affects those calls to server utilities made from within the rpasInstall process and does not affect calls to server utilities made from the command line or as part of a batch job. The syntax of the property is:

-Drpas.maxProcesses=###, where ### is the number of sub-processes the RPAS server utility should attempt to run in parallel. The number of processes to use should be determined by the administrator of the hardware system based on the physical number of processors available and the amount of load that is acceptable for the rpasInstall process to place on the system.

Before You Begin

Before starting the RPAS Installer, the following software must be installed on your system:

- Java 1.7u91
- Unzip utility

Running the RPAS Installer

Perform the following procedure to run the RPAS Installer.

1. Locate and extract RPAS-14.1.2-unix.zip into the current directory, which is referred to in this document as **[RPAS_Installer]**.

   Note: This product's installer includes Ant. If Ant is already installed on your system and is version 1.6.5 or earlier, then you must run unset ANT_HOME on the command line to ensure that the installer uses the included version. The unset ANT_HOME command must be run before running ./install.sh.

2. Begin the Installer by changing to the **[RPAS_Installer]** directory and running the following command:

   ```
   ./install.sh
   ```

Note: The command must be run with the preceding period and slash (./).

If this process is being run on an X-Windows emulator (such as Exceed), a graphical user interface (GUI) to the Installer opens. If you are running in console mode through a terminal emulator, a text interface to the Installer opens.

In both cases, the requested information is identical, but displayed differently. In the GUI, a checkbox may appear to indicate whether you want a component installed. In text mode, you are prompted for a response of either *Yes* or *No*.

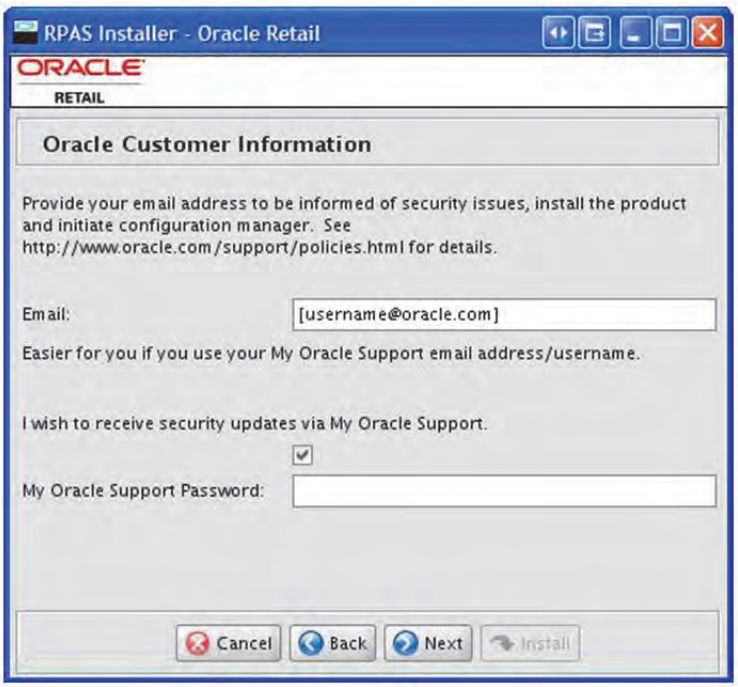
Note: In text mode, the default value appears in square brackets. To use the default value and continue, press **Enter**. If you want to use a different value, enter the new value. When prompted to create a directory, respond with *Yes* and press **Enter**.

3. The RPAS Installer Window opens and displays the components that can be installed to your system. Click **Next**.

Figure 3–1 RPAS Installer Window

4. The Oracle Customer Information Window opens.

Figure 3–2 Oracle Customer Information Window



If you want to receive e-mails from My Oracle Support about security updates, enter your e-mail address and My Oracle Support password and ensure that the check box is selected. Click **Next** to continue.

The security updates are provided through Oracle Configuration Manager (OCM). The Oracle Retail OCM collector is included in the installer and is shown in Figure 3–2. The collector only needs to be installed once per ORACLE_HOME, WAS_HOME, or installation root directory. After the initial installation, the OCM collector automatically performs self-updates.

For more information about Oracle Retail OCM, refer to the following guide:

Oracle Configuration Manager Installer Guide (Note ID: 1071030.1)

This guide describes the procedures and interface of the Oracle Retail OCM collector that is a part of Oracle Retail full releases.

This document is available through My Oracle Support. Access My Oracle Support at the following URL:

`https://support.oracle.com`

The OCM documentation is located at the following URL:

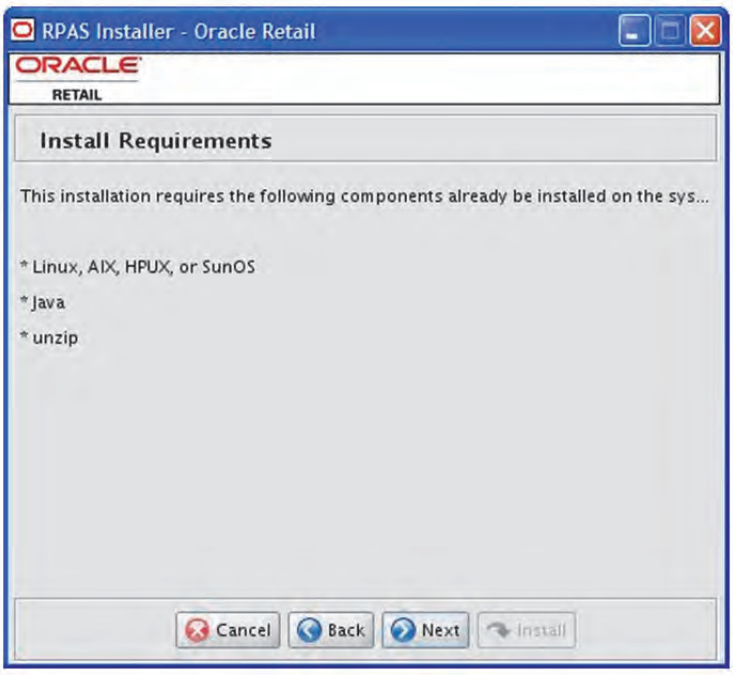
`http://www.oracle.com/technology/documentation/ocm.html`

Note: If you select to receive security updates but do not provide e-mail and password information or lack an internet connection, additional windows appear. For more information about these windows, refer to the *Oracle Configuration Manager Installer Guide*.

---

5. The Install Requirements Window opens.

Figure 3–3 Install Requirements Window

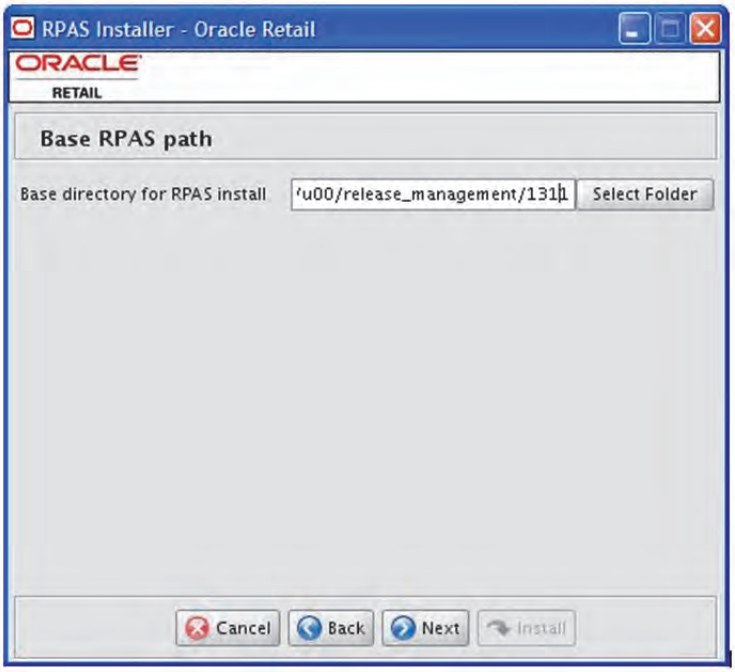


This window displays the software required to complete this installation. You should already have installed this software on your system. If you have not installed these items, perform the necessary installations before continuing.

Click **Next** to continue.

6. The Base RPAS Path Window opens.

Figure 3–4 Base RPAS Path Window



Enter the path where the RPAS Server and components will be installed, and click **Next**. This path is used when creating your domains, should you choose to create them.

Note: Ensure that the installation path is located outside of the installer directory **[RPAS_Installer]**.

If this directory does not exist, the Installer prompts you to create it.

---

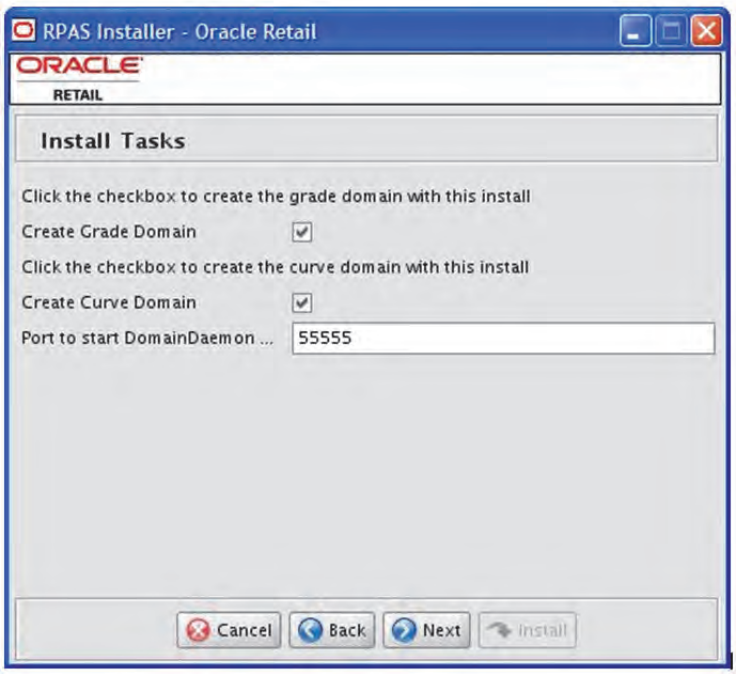
7. The RPAS Installations Paths window opens.

   Enter the following path information and click **Next**:

Field	Description
RPAS Server path	Enter the target directory for your RPAS Server.
RPAS Tools path	Enter the target directory for your RPAS Configuration Tools.
Grade Config path	Enter the target directory for your Grade configuration.
Curve Config path	Enter the target directory for your Curve configuration.
Directory for Retail login script	Enter the target path where the `retaillogin.ksh` file will be created on your system.

8. The Install Tasks Window opens.

Figure 3–5 Install Tasks Window



Select the options to be performed by the RPAS Installer, enter the DomainDaemon port number, and click **Next**.

- To create the domains for Grade and Curve, select the appropriate options on the Install tasks window. These options are selected by default, meaning that they will be created.
- Enter the port where the RPAS DomainDaemon will run. This port needs to be configured for use with the RPAS Clients.

  For the RPAS Classic Client, this is done with the EConfigure utility as documented in Chapter 6, "Installing and Configuring the RPAS Classic Client".

  For the RPAS Fusion Client, this is done during the RPAS Fusion Client installation as documented in Chapter 5, "Installing the RPAS Fusion Client".

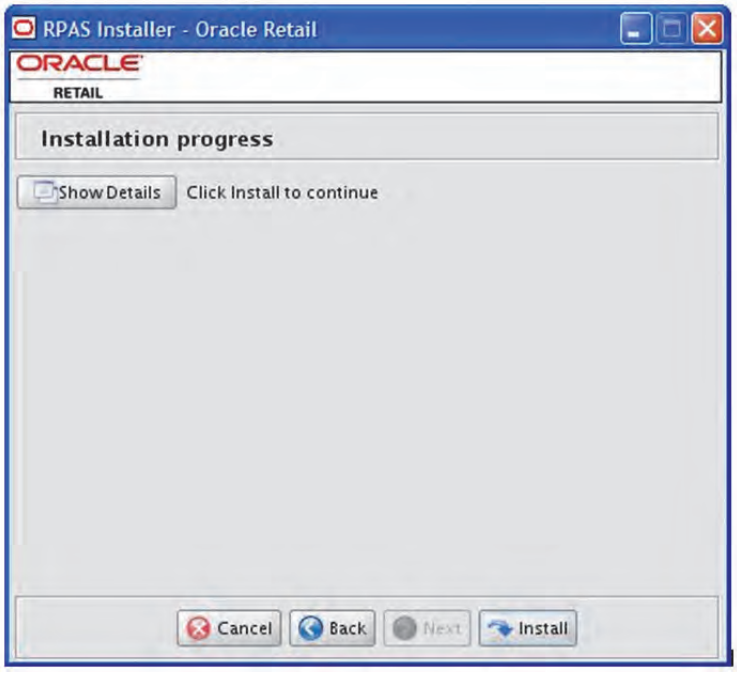
The Installer will validate that this port is not in use. The DomainDaemon will not be running at the end of this installation process, but can be started by using the startrpas alias created in the environment setup script.

Note: If you choose to create domains, they are created in a directory called domains under the **Base directory path** you defined on the Base RPAS path window.

---

9. The Installation Progress Window opens. to start the installation, click **Install**.

Figure 3–6 Installation Progress Window



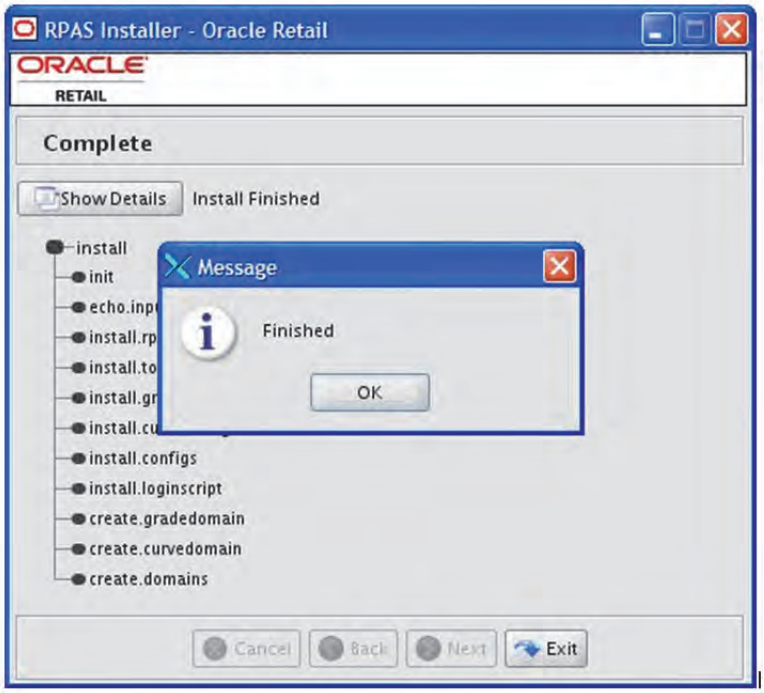
To display the progress of the components and tasks being performed by the Installer, click **Show Details**. You can view the detailed mode at any time during or after the installation.

Note: The installation process can vary depending on your environment. If you chose to create the domains, installation time might take 10 to 30 minutes depending on server.

10. When the installation process is complete, the Complete Window opens with a *Finished* Message dialog box.

    Click **OK** to close the dialog box.

Figure 3–7 Complete Window



11. To view the installation details, click **Show Details**. The window displays two tabs: the Output tab and the Error tab. It is recommended that you review these tabs for any issues that may have occurred during the installation process.

    If you want to view the log again at a later date, a text copy was saved in the directory **[RPAS_Installer]**. The log file will be named based on the product, platform, and a timestamp, followed by the `.log` extension.

12. Click **Exit** to close the Installer.

Environment Variable Setup Script

To begin using RPAS, run the `retaillogin.ksh` script. The script is located in the root of the base directory where RPAS was installed unless the default was overwritten when specifying directory paths.

Source the script from inside the directory where the script is located:

```
. ./retaillogin.ksh
```

Or, include the full path after the period and space ".":

```
. /<base_directory>/retaillogin.ksh
```

> **Note:** The preceding period and space (".") must be included at the beginning of the command when running the script.
>
> Include this path and script in the `.profile` in your home directory (`~/.profile`) if you want to have this environment setup script run during login.

This script will set up environment variables, such as **RPAS_HOME** and **RIDE_HOME**, which are required for RPAS to run properly.

Installing ODBC Server and Client Components

After the RPAS Server has been installed and the **RPAS_HOME** environment variable has been set, the RPAS ODBC Server can be installed. The ODBC Server is required only if you have ODBC or JDBC applications that use the RPAS domain as data source, or if you want an SQL interface to the RPAS domain.

To install the ODBC Server, change directory to **RPAS_HOME/odbc** and run the shell script `customInstall.sh`. Although the installation script `customInstall.sh` works for non-root users, it is strongly suggested that the script be run by a root user so that the RPAS ODBC Agent can utilize the OS logon authentication method.

For information about the ODBC server configuration, refer to either the Classic Client or Fusion Client version of the *Oracle Retail Predictive Application Server Administration Guide*.

Both 32-bit and 64-bit ODBC Clients are available. They are delivered in zip files named `odbcclient32.tar.zip` and `odbcclient64.tar.zip` respectively. To install, copy the appropriate zip file to your preferred location. Unzip and untar the file. For more information about the ODBC Client configuration, refer to either the Classic Client or Fusion Client version of the *Oracle Retail Predictive Application Server Administration Guide*.

Installing JDBC Client

Perform the following procedure to install the JDBC client driver.

Unzip and untar the `jdbcclient.tar.zip` from the **RPAS_HOME** directory.

For additional information on the JDBC Server, refer to either the Classic Client or Fusion Client version of the *Oracle Retail Predictive Application Server Administration Guide*.

Determine the Path for the Domains

Perform the following procedure to Determine the Path for the Domains.

1. Determine the locations of the domains to be installed.

   > **Note:** Domain paths cannot contain spaces. In addition, symbolic links cannot be used for domain paths.

2. Create a directory at the root of the domain to be installed.

DomainDaemon

The RPAS DomainDaemon is the process that must be running on the server for a user to log into an RPAS domain. Before beginning the installation process, a port was specified where the DomainDaemon will run.

You can start the DomainDaemon manually. Instructions for the DomainDaemon are included in the Classic Client and Fusion Client versions of the *Oracle Retail Predictive Application Server Administration Guide*.

4

Installing on a Windows Environment

This chapter describes how to install RPAS on a Windows environment.

RPAS Server and Tools Installation on Windows

Note: You must install the RPAS Server on Windows to use the RPAS Configuration Tools on Windows. The RPAS Configuration Tools application is supported on Windows 7 and 8.1.

For the purposes of this section, a slash "/" is used to delineate directories and files in paths. Users in a Windows Command Prompt environment need to either use a backslash "\" as the delineation character or use double quotes around paths.

Note: Paths on Windows are case-sensitive when using Cygwin.

Extracting the RPAS Package

Unzip the `RPAS-14.1.2-windows.zip` to a newly created directory on the Windows machine. The `RPAS-14.1.2-windows.zip` contains all the RPAS components.

Once extracted, the following directories appear:

Directory	Description
ClassicClient	This directory contains the `setup.exe` used to install the RPAS Classic Client.
Curve	This directory contains the Curve base configuration file provided with RPAS.
Grade	This directory contains the Grade base configuration file provided with RPAS.
ODBC	This directory contains the JDBC and ODBC client drivers.
RPAS	This directory will be referred to later in this document as **RPAS_HOME**.
Tools	This directory will be referred to later in this document as **RIDE_HOME**.
Translations	This folder contains the files for the various languages supported by RPAS.
Web	This directory contains the files required for an RPAS Web deployment.

Java Environment

During the Java installation, a directory is created to store the Java software. This directory is referred to later in this document as **JAVA_HOME**.

Install Cygwin

Cygwin is required if the RPAS Server is to be installed on Windows operating systems. You can find more information about downloading this product at:

`http://www.cygwin.com`.

For installation guidance, refer to the following note on My Oracle Support (`https://support.oracle.com`):

Cygwin Setup Guide for RPAS (Note 1333398.1)

This document details necessary options you should select when installing Cygwin to make it function properly with RPAS.

Cygwin Path Settings

To export PATH through Cygwin, use colons (:) and to export RPAS_JAVA_CLASSPATH through Cygwin, use semicolons (;). If you are using Java in Cygwin, use semi-colons to seperate Java class paths. Do not use colons because `Java.exe` runs on Windows and it expects a semi-colon. For this instance, using a colon is incorrect as it is already used in the Windows file paths. Cygwin is a UNIX like environment and as the command line interface for Windows it expects colons for path settings.

> **Note:** To setup this variable, refer to Step 6 in Creating the Required Environment Variables.

Example 4–1 Cygwin Path

```
export PATH=/usr/bin:/usr/local/bin:/cygdrive/c/java/32bit/jdk1.7.0_20/bin
export RPAS_JAVA_CLASSPATH= "c:/rpas_home/13.3/rpas/applib/aaijni.jar;
                             c:/rpas_home/13.3/rpas/applib/aaiReplenOpt.jar"
```

Determine the Path for the Domains

Perform the following procedure to determine the path for the domains.

1. Determine the locations of the domains to be installed.

> **Note:** Domain paths cannot contain spaces. In addition, symbolic links cannot be used for domain paths.

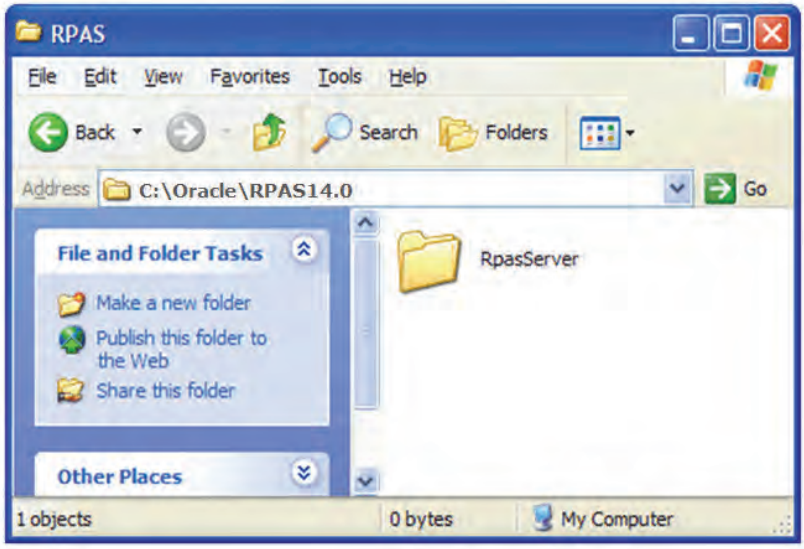
2. Create a directory at the root of the domain to be installed.

Installing the RPAS Server

The following procedures provide information about creating the necessary folders on your Windows PC and copying the RPAS Server components to them.

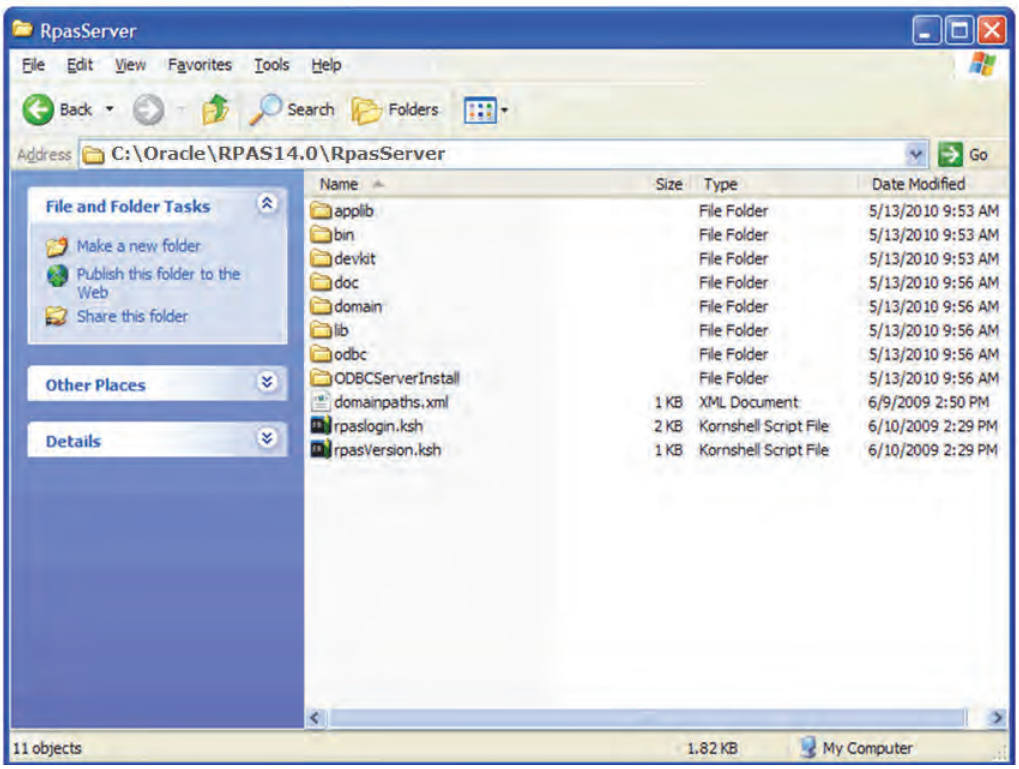
1. On your C drive, create a folder named **Oracle**.
2. Open the **Oracle** folder and create a folder named **RPAS14.1.2**.
3. Open the **RPAS14.1.2** folder and create a folder named **RpasServer**.

Figure 4–1 Example of RpasServer Folder Path



4. Copy all files and folders from the **RPAS** folder where you extracted the Media Pack to the **C:\Oracle\RPAS14.1.2\RpasServer** folder.

Figure 4–2 RpasServer Folder with All RPAS Server Components



This location is referred to as **RPAS_HOME**. An environment variable is defined on your Windows PC to point to this location so that RPAS will function correctly. Refer to Creating the Required Environment Variables for information on creating

the necessary RPAS variables.

Note: When installing on Windows 7, update the permissions for **$RPAS_HOME/bin** directory to include run permissions.

In a Cygwin session, enter the following commands:

```
cd $RPAS_HOME
chmod +x -R bin
```

Where `RPAS_HOME` is set to the correct directory and is in the proper Cygwin format.

Microsoft 2010 Runtime Libraries

Ensure that these Microsoft 2010 Runtime Libraries are installed to enable RPAS Configuration Tools and other RPAS utilities:

- MSVCR100.dll
- MSVCP100.dll

Download these libraries from this link:

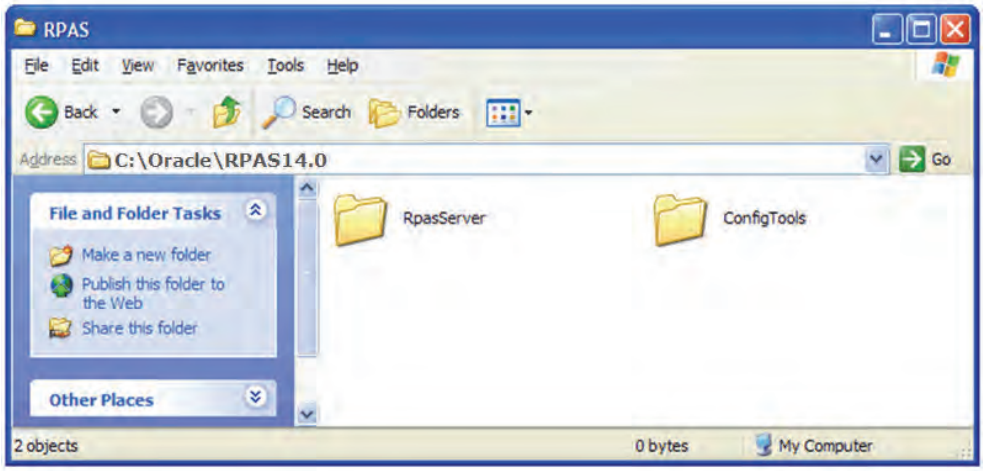
`http://www.microsoft.com/en-us/download/details.aspx?id=5555`

Installing Configuration Tools

The following procedures provide information about creating the necessary folders on your Windows PC and copying the Configuration Tools components to them.

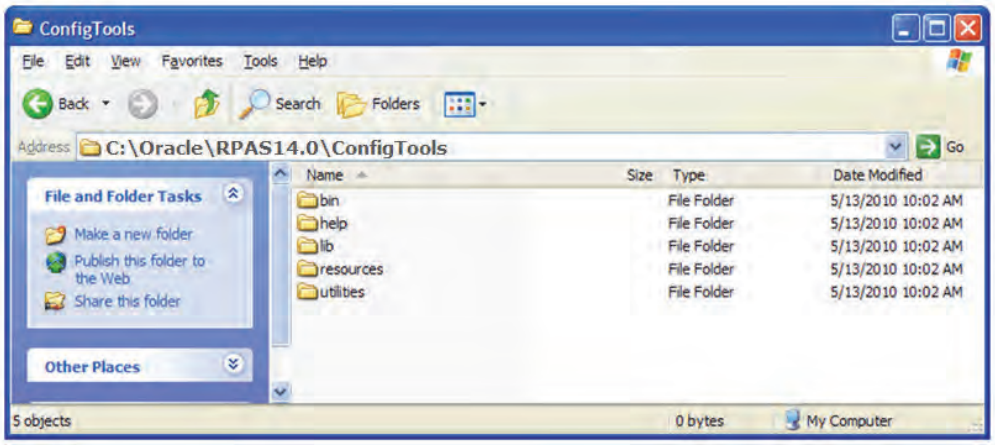
1. Using Windows Explorer, navigate to your C drive, create a folder named **Oracle\RPAS14.1.2**, which you created in the Installing the RPAS Server section.
2. Create a folder named **ConfigTools**.

Figure 4–3 Example of ConfigTools Folder Path



3. Copy all files and folders from the **CDROM\Tools** folder where you extracted the Media Pack to the **C:\Oracle\RPAS14.1.2\ConfigTools** folder.

Figure 4–4 ConfigTools Folder with All Configuration Tools Components



This location is referred to as **RIDE_HOME**. An environment variable will be defined on your Windows PC to point to this location so that RPAS will function correctly. Refer to the Creating the Required Environment Variables section for information on creating the necessary RPAS variables.

Note: If you plan to use the Curve or Grade functionality, ensure that you copy the **plugin** folders to the **ConfigTools** folder. For instructions, refer to the section, Building the Domain on your Windows PC.

Install ODBC or JDBC Client Components (Optional)

ODBC Client Driver

Perform the following procedure to install the ODBC client driver.

- To install the ODBC client software, run `setup.exe` from the ODBC\ODBCClient directory where you extracted the `RPAS-14.1.2-windows.zip` file.
- Follow the installation wizard to proceed to the Oracle RPAS ODBC Driver Setup window. Enter the destination folder if the default values need to be modified.
- Click **Next**. The Data Source Configuration window opens. If the default values need to be modified, enter the data source name, description, service host name, service port, and service data source name.
- Follow the rest of the installation wizard to finish the installation process.

JDBC Client Driver

Perform the following procedure to install the JDBC client driver.

1. Unzip and untar the `jdbcclient.tar.zip` from the ODBC/JDBC Client directory where you extracted the `RPAS-14.1.2-windows.zip` file to a target destination directory.

For additional information on the ODBC Server, refer to either the Classic Client or Fusion Client version of the *Oracle Retail Predictive Application Server Administration Guide*.

Install ODBC Server Components (Optional)

Perform the following procedure to install the ODBC server:

1. Run `setup.exe` from **Rpas/ODBCServerInstall/iwinnt** folder where you extracted the `RPAS-14.1.2-windows.zip` file.
2. Follow the installation wizard to proceed to the Server Setup window. In this window, enter the destination folder path.
3. The Server Configuration window opens. If the default values need to be modified, enter the agent service name, agent service port, and the Windows username used to administer this installation of the server.
4. Proceed to the next window, Service Configuration. In this window, enter the service name and service port if the default values need to be modified.
5. Follow the rest of the installation wizard to finish the installation process.

> **Note:** If the installation fails on Windows 7, try installing again in compatibility mode. To do this, perform the following steps:
>
> Right-click the `setup.exe` file and select the **Troubleshoot compatibility** option. Then, select the **Try recommended settings** option and click **Start the program**. Follow the installation wizard to complete the installation process.

For additional information on the ODBC Server, refer to either the Classic Client or Fusion Client version of the *Oracle Retail Predictive Application Server Administration Guide*.

Creating Start Menu Shortcuts to RPAS Applications and Utilities

This section provides information on creating shortcuts to these applications:

- Configuration Tools
- Configuration Converter

Procedure to Create Start Menu Shortcuts to RPAS Applications and Utilities

Perform the following procedure to create start menu shortcuts to RPAS applications and utilities

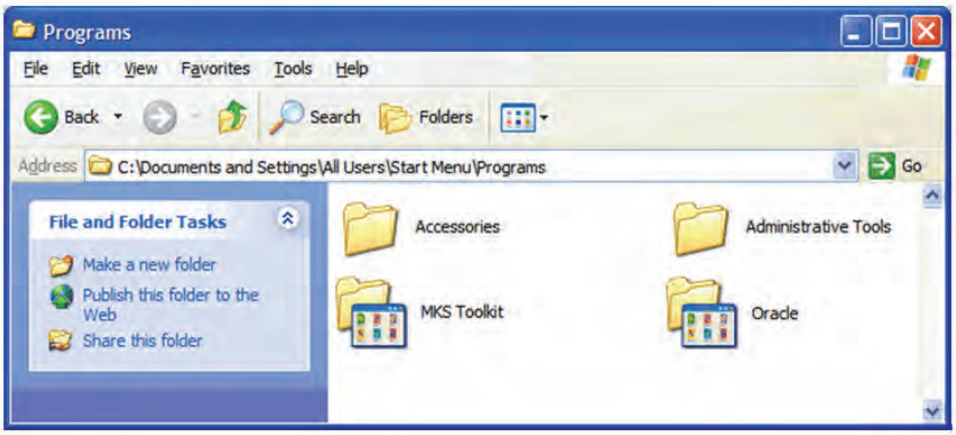
1. Open Windows Explorer and navigate to the appropriate path for your operating system.

Operating System	Path
Windows 7	C:\ProgramData\Microsoft\Windows\Start Menu\Programs

   The Programs window displays all applications and shortcuts available to all users accessing the PC.

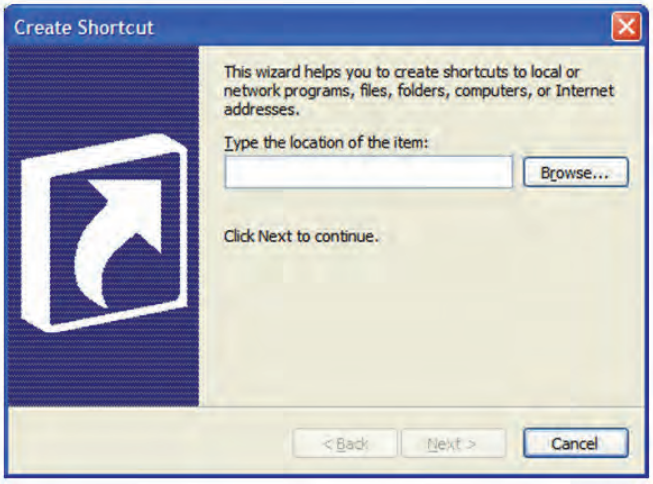
2. Right-click the window and select **New – Folder**. Name this folder **Oracle**.

Figure 4–5 Programs Windows with Oracle Folder



3. Double-click the **Oracle** folder. The folder opens in Windows Explorer.
4. Right-click the Oracle window and select **New – Folder**. Name the folder **RPAS 14.1.2**.
5. Create a shortcut to Configuration Tools:
   a. Double-click the **RPAS 14.1.2** folder, right-click in the folder window, and select **Shortcut**. The Create Shortcut Wizard Dialog Box opens.

Figure 4–6 Create Shortcut Wizard Dialog Box



   b. Click **Browse** and navigate to the **Oracle\RPAS14.1.2\ConfigTools\bin** folder on your C drive.
   c. Select `ConfigTools.exe` and click **OK**. The selected path appears in the Create Shortcut wizard.
   d. Click **Next**. The Select a Title for the Program wizard window opens. By default, this window displays the file name and extension selected.
   e. In the text field, enter **Configuration Tools** as the shortcut name and click **Finish**. The shortcut wizard closes and the Oracle window displays the shortcut to the Configuration Tools.

6. Create a **Utilities** folder in the Oracle window. Right-click the window and select **New – Folder**. Name this folder **Utilities**.
7. Double-click the **Utilities** folder. The folder opens in Windows Explorer.
8. Create a shortcut for the Configuration Converter:
   a. Right-click the **Utilities** folder window and select **Shortcut**. The Create Shortcut wizard dialog box opens.
   b. Click **Browse** and navigate to the **Oracle\RPAS14.1.2\ConfigTools\utilities** folder on your C drive.
   c. Select `RpasConverter.exe` and click **OK**. The selected path appears in the Create Shortcut wizard.
   d. Click **Next**. The Select a Title for the Program wizard window opens.
   e. In the text field, enter **Configuration Converter – g** as the shortcut name and click **Finish**. The **–g** after the name is required to display the Windows interface for the converter. The shortcut wizard closes and the Oracle window displays the shortcut to the Installer.
9. Verify your shortcuts appear in the Start menu. From the Start menu, select **All Programs – Oracle – RPAS 14.1.2.** The Configuration Tools and Installer shortcuts should appear. Select the **Utilities** folder and verify that the Configuration Converter shortcut appears.

   Now that the necessary files and shortcuts are defined, you need to create the necessary environment variables in order to open the applications

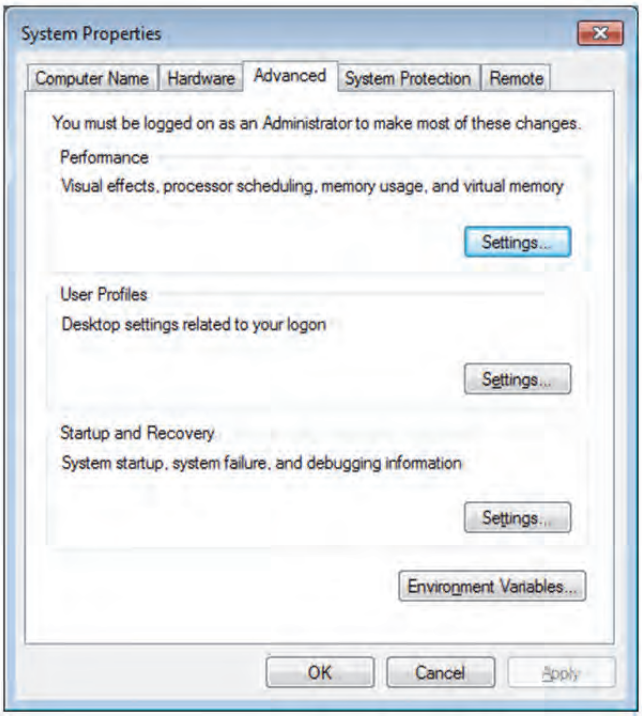
Creating the Required Environment Variables

The following steps outline the process to follow and environment variables required to support the RPAS installation and domain install.

1. From the Control Panel, open the System window.
   a. If your system is using Category view, from the Windows 7 Start menu, go to **Control Panel**, select **System and Security**, select **System**, and then select **Advanced System Settings**.

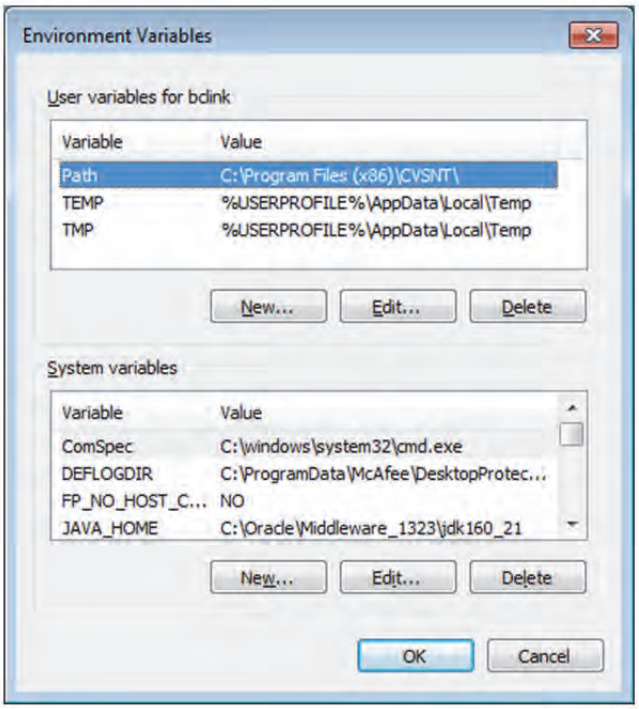
      The System Properties window opens on the Advanced tab.
   b. Select the **Advanced** tab.
   c. At the bottom of the window, click **Environment Variables**.

Figure 4–7 System Properties - Advanced Tab



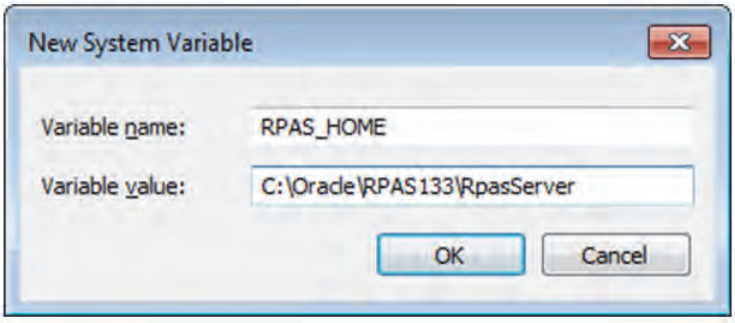
d. The Environment Variables window opens.

Figure 4–8 Environment Variables



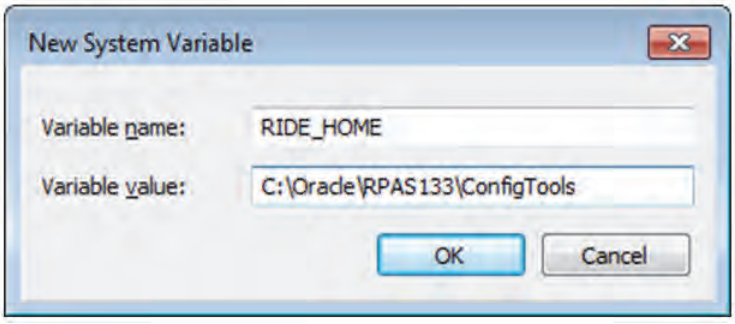
2. Create the **RPAS_HOME** environment variable.
   a. Under the System variables box, click **New**. The New System Variable dialog box opens.
   b. Enter **RPAS_HOME** in the **Variable** name field.
   c. Enter the path the **RPAS Server** folder in the **Variable value** field.

Figure 4–9 Example of RPAS_HOME Variable



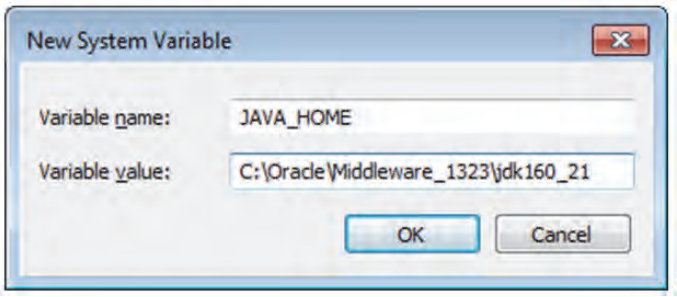
   d. Click **OK**. **RPAS_HOME** now appears in the Variable name box.
3. Create the **RIDE_HOME** environment variable.
   a. Under the System variables box, click **New**. The New System Variable dialog box opens.
   b. Enter **RIDE_HOME** in the **Variable name** field.
   c. Enter the path the **ConfigTools** folder in the **Variable value** field.

Figure 4–10 Example of RIDE_HOME Variable



   d. Click **OK**. **RIDE_HOME** now appears in the Variable name box.
4. Create the **JAVA_HOME** environment variable.
   a. Under the System variables box, click **New**. The New System Variable dialog box opens.
   b. Enter **JAVA_HOME** in the **Variable name** field.
   c. Enter the path the **Java** folder under Program Files in the **Variable value** field.

Figure 4–11 Example of JAVA_HOME Variable



d. Click **OK**. **JAVA_HOME** now appears in the Variable name box.

5. Update the Path variable.

   a. Under the System variables section, select the **Path** environment variable and click **Edit**.

   b. Insert the complete paths for **RPAS_HOME, RIDE_HOME**, and **JAVA_HOME** as shown:

```
%RPAS_HOME%\bin;%RPAS_HOME%\applib;%RPAS_HOME%\lib;

%RIDE_HOME%\bin;%RIDE_HOME%\lib;

%JAVA_HOME%\bin;%JAVA_HOME%\bin\client;%JAVA_HOME%\lib;

%JAVA_HOME%\jre\bin\client
```

   Note: Remember to separate all path statements with semicolons (;).

   c. Select **OK** to save your changes.

6. Update the **RPAS_JAVA_CLASSPATH** environment variable.

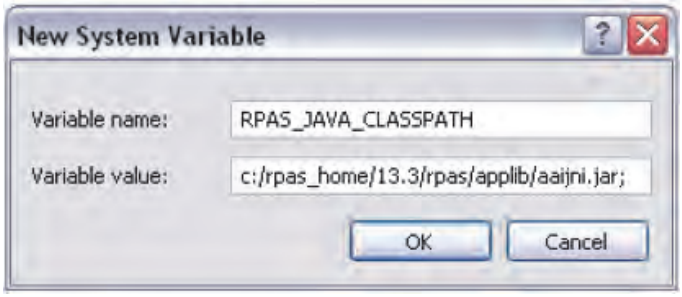
   Note: Not all RPAS solutions make use of the RPAS_JAVA_CLASSPATH. Refer to the specific product implementation guide for information on whether that solution requires the RPAS_JAVA_CLASSPATH variable and what value should be provided.

   a. Under the System variables box, click **New**. The New System Variable dialog box opens.

   b. Enter **RPAS_JAVA_CLASSPATH** in the **Variable name** field.

   c. Enter the path the **Java** folder under Program Files in the **Variable value** field.

Figure 4–12 Example of RPAS_JAVA_CLASSPATH Variable



d. Click **OK**. **RPAS_JAVA_CLASSPATH** now appears in the Variable name box.

7. Close all open windows.

Create a Global Domain Configuration Directory (Optional)

If installing a Global Domain environment, an xml file may be created to determine how the domains will be partitioned and the label of each domain. The following is an example of the structure of the `globaldomainconfig.xml` file. The items in Example 4–2 are as follows:

Item	Description
Path	The location of the root of the domain.
Partitiondim	The partition dimension. Using the following example, pgrp (Group) is the dimension in which the local domains are partitioned. There can only be one partition dimension.
Subpath	The path and name of the local (sub-domain) that contains a specific partition position. ldom+# is the default name given by RPAS to local domains.
Subposition	The position from the partition dimension that is located in the local domain. For example, ldom0 includes all product positions at or below pgrp 1100.

Example 4–2 File Structure

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<rpas>
    <globaldomain>
        <path>/Domains/RDF133/D01</path>
        <partitiondim>pgrp</partitiondim>
        <subdomain>
            <subpath>/Domains/RDF133/ldom0</subpath>
            <subpositions>1100</subpositions>
        </subdomain>
        <subdomain>
            <subpath>/Domains/RDF133/ldom1</subpath>
            <subpositions>1300</subpositions>
        </subdomain>
        <subdomain>
            <subpath>/Domains/RDF133/ldom2</subpath>
            <subpositions>2500</subpositions>
        </subdomain>
    </globaldomain>
</rpas>
```

Configure the RPAS Clients to Use the Domain

The RPAS Clients must be configured to point to the newly created domains.

For instructions on how to configure the RPAS Clients, refer to either the Classic Client or Fusion Client version of the *Oracle Retail Predictive Application Server Administration Guide*.

Using Multiple Versions of RPAS on the Same Windows Machine

If you have multiple versions of RPAS installed on your PC, it is important to note that the environment variables will reference RPAS 14.1.2 after the installation process is complete.

Note: Previously set environment variables for other versions or installations of RPAS will still exist in the Path System variable, but Windows uses the first set of variables defined in the path, which is where the installation process places them.

To switch to a different version of RPAS that is installed on your machine, you will need to manually update the environment variables each time you want to switch. You can either insert the path to the version you want to use and leave the path to 14.1.2, or delete the path and either reinstall the 14.1.2 components or manually reinsert the paths when you want to revert to 14.1.2.

Base Configuration Installation

There are two base configurations available with the RPAS archive that can be used to build a domain. These configurations are initially stored in the following folders that were created in the **CDROM** folder when you extracted the `RPAS-14.1.2-windows.zip` file in the location where you downloaded the RPAS Media Pack.

- **Grade** – A clustering tool that provides insight into how various parts of a retailer's operations can be grouped together.
- **Curve** – A profile generation tool used to produce ratios (profiles) from historical data at user-specified intersections.

The following section describes how to use these configurations to build a domain.

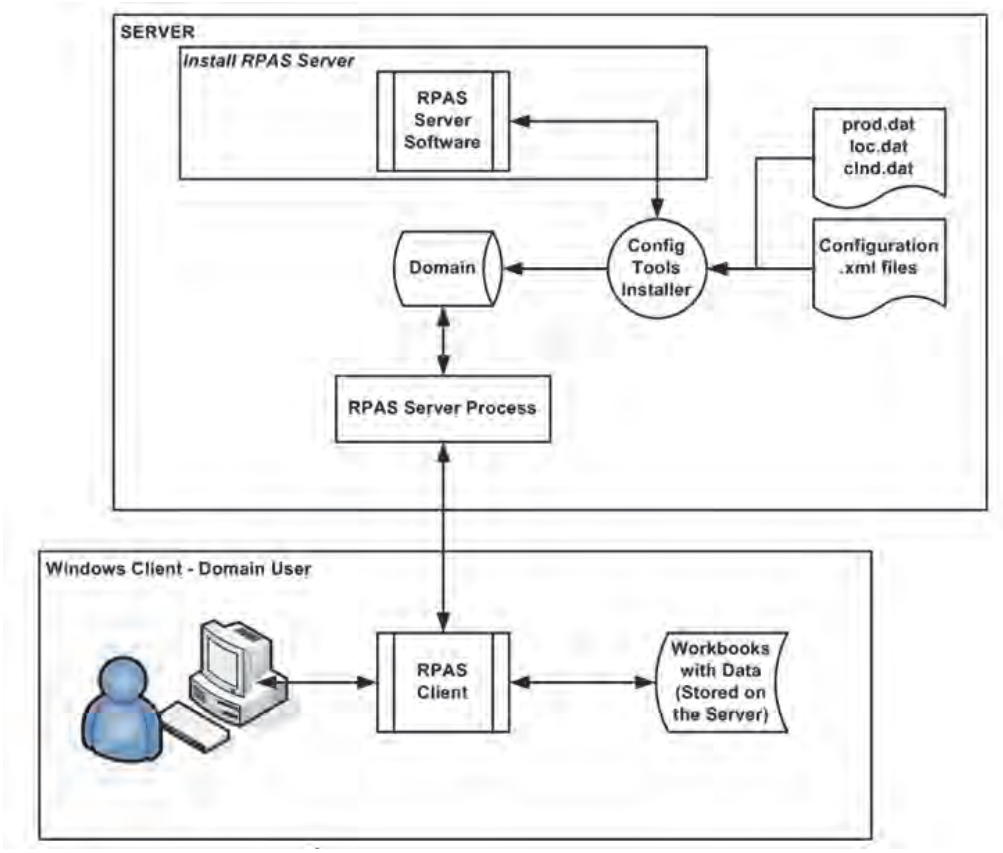
Build a Domain Process Overview

Figure 4–13 shows an overview of the steps involved in using a configuration to build and an RPAS domain.

This section describes the steps illustrated in Figure 4–13:

1. Verify the Environment Variable Settings
2. Setting Up Base Configuration Files
3. Building the Domain on your Windows PC
4. Build the Domain
5. Start the RPAS Server (DomainDaemon)

Figure 4–13 Build a Domain Process Diagram



Verify the Environment Variable Settings

Prior to beginning the domain build process you should have installed RPAS and the Configuration Tools on your server. During that process, you should have set up the necessary environment variables for RPAS and the Configuration Tools.

Perform the following steps to verify that environment variables have been successfully configured.

Open a `Cygwin zsh` window. Use the following commands to verify your environment settings:

```
echo $RPAS_HOME
echo $RIDE_HOME
echo $JAVA_HOME
echo $PATH
```

> **Note:** The path for the **RPAS_HOME** variable may change from release to release.

If you make any changes to the environment variable settings, remember to exit and restart your UNIX session in order to run your `.profile` and make the changes effective. This step is very important before you continue to the remaining steps.

Note: The paths for your **RIDE_HOME** and **RPAS_HOME** variables cannot have spaces in them, unless short file naming conventions are utilized. Given this restriction, do not place your Tools build, Tool Configurations or RPAS installation under Program Files or My Documents. If you do, define all RPAS related environment variables using short (8dot3) file names.

Setting Up Base Configuration Files

Locate and copy/move the desired configuration zip file to a location on your machine. For the purposes of these instructions assume that location is called:

C:/root/testenv/<Configuration>

The following subfolders will be created in: **C:/root/testenv/<Configuration>**:

Subfolder	Description
data/ – hierarchy and sample data files	This path is used in conjunction with the `-in` [input] option of the `rpasInstall` command.
configuration/<Configuration_Name>	Configuration files for use in building the domains where <Configuration_Name> will be Sample_Configuration, Grade, or Curve. Do not change the directory name for the configuration or alter the contents in any way.
scripts	Scripts used to complete the domain build process.
resources	Contains the plug-ins for the Configuration Tools. When this resource directory is supplied, it must be copied in its entirety to the **RIDE_HOME** location. This must be done in order for the Configuration Tools to build domains.

Building the Domain on your Windows PC

This section provides instructions for how to create a domain from the base configurations.

Because building an RPAS domain on Windows is currently a manual process, the person building the domain should be skilled in administering UNIX or Windows servers and should have scripting skills.

Note: The Configuration Tools are supported on all platforms (HP-UX, AIX, and Windows); however, they require Java 1.7. Make sure that the server you will be using has this version of Java.

1. Create a **testenv** (test environment) folder on your C drive.
2. Copy the Curve and Grade **plugin** folders to the **ConfigTools\resources\plugins** folder. These Curve and Grade **plugin** folders are located in the folder where you extracted the `RPAS-14.1.2-windows.zip` file to the **testenv** folder:

   testenv\Curve\resources\plugins

   testenv\Grade\resources\plugins

Sample Data Files

The domain build process requires the following data files to be available:

- prod.dat
- loc.dat
- clnd.dat
- input data files for measures (<measure>.ovr)

These files should be located in the **C:/root/testenv/<Configuration>/data** directory. This directory path will be used during the domain build process as the input directory.

Domain Environment Setup

The path that the domain will be created must exist prior to running the domain build process.

For the domain, manually create the directory structure:

```
C:/root/testenv/domain
```

The name of the domain will automatically be created under the domain path based on the configuration name. So, for the previously listed domain path, the full path to the domain will be as follows once the domain build process is completed.

```
C:/root/testenv/domain/<Configuration>
```

Build the Domain

Use the Tools Installer, the rpasInstall script, to build the domain. This executable is located in bin directory of your Tools installation. There are different scripts to run based on which configuration is being used to build a domain.

For more information on the Tools Installer and the specific options available when using the rpasInstall command, refer to the *Oracle Retail Predictive Application Server Configuration Tools User Guide*.

> **Note:** The `rpasInstall` script only loads the hierarchy files and builds the domain. It does not load any measure data. The hierarchy files are copied to the /input/processed directory of the domain and appended with a time-date stamp.

Building the Grade Domain

Perform this procedure to build the domain for Grade.

1. Copy the **plugins** folder from the Grade/resources directory and paste it in the **RIDE_HOME/resources** directory.
2. Run `execPluginTask.sh` for Grade, which creates the cluster hierarchy files (`clsh.dat` and `grch.dat`) in the directory <path to the data files>:

   ```
   execPluginTask.sh Grade:com.retek.labs.grade.plugin.GradeDataGenerator
   <path to the configuration>/Grade/Grade.xml <path to the data files>
   ```

   > **Note:** The <path to the data files> is the argument for `-in` of rpasInstall.

Note: To build a Grade workbook, a junk position is required in the cluster hierarchy. Therefore, a junk position must exist in the `clsh.dat`. as shown in the following example:

1	1 Cluster01
2	2 Cluster02
3	3 Cluster03
4	4 Cluster04
5	5 Cluster05
Junk	Junk Cluster

3. If performing a full installation, omit this step and continue to Step 4

   When performing a patch installation, manually copy the cluster hierarchy files (`clsh.dat` and `grch.dat` - generated in Step 2), from directory <path to the data files> to directory <path to the domain>/Grade/input:

```
for PatchFile in \
  clsh.dat \
  grch.dat
do
  orgFile=<path to the data files>/${PatchFile}
  destFile=<path to the domain>/Grade/input/${PatchFile}

  if [[ -f "${orgFile}" ]]; then
    cp -f "${orgFile}" "${destFile}"
  fi
done
```

4. Enter the following command to build a domain for the Grade configuration:

   ```
   rpasInstall -fullinstall -dh <path to the domain> -cn Grade -ch <path to the configuration> -in <path to the data files> -log <path to the location and name of the installation log> -rf AppFunctions –rf ClusterEngine –p pgrp
   ```

5. After the domain installation has completed, the sales data must be loaded into the domain using the loadmeasure utility. Open a command prompt from the master domain (/Grade) and type the following commands:

   ```
   loadmeasure -d . -measure dpos
   ```

   ```
   loadmeasure -d . -measure rsal
   ```

   ```
   loadmeasure -d . -measure csal
   ```

   ```
   loadmeasure -d . -measure psal
   ```

6. Open a command prompt from the local domain (/Grade/ldom0) and type the following command:

   ```
   mace -d . -run -group common_batch
   ```

   Repeat this step for each of the remaining local domains (/Grade/ldom1, /Grade/ldom2).

Building the Curve Domain

Perform this procedure to build the domain for Curve.

1. Copy the **plugins** folder from the Curve/resources directory and paste it in the **RIDE_HOME/resources** directory.

2. Run `execPluginTask.sh` for Curve, which creates Curve data files in the directory <path to the data files>

```
execPluginTask.sh
Curve:com.retek.labs.curve.plugin.installer.InstallParameterDataGenerat
ion <path to the configuration>/Curve/Curve.xml <path to the data
files>
```

Note: The <path to the data files> is the argument for `-in` of rpasInstall.

Note: The data files generated in this step include:

dataintxl.ovr
flprofxl.ovr
pgdatatmp.ovr
pgdefapmthtmp.ovr
pgdefmasktmp.ovr
pgmthtwtmp.ovr
pgnvtmp.ovr
prfaggintxl.ovr
prfaprvintxl.ovr
prfintxl.ovr
proftypexl.ovr
slprofxl.ovr
strintxl.ovr.

---

3. If performing a full installation, omit this step and continue to Step 4

   When performing a patch installation, manually copy the data files generated in Step 2), from directory <path to the data files> to directory <path to the domain>/Curve/input:

```
for PatchFile in \
  dataintxl.ovr \
  flprofxl.ovr \
  pgdatatmp.ovr \
  pgdefapmthtmp.ovr \
  pgdefmasktmp.ovr \
  pgmthtwtmp.ovr \
  pgnvtmp.ovr \
  prfaggintxl.ovr \
  prfaprvintxl.ovr \
  prfintxl.ovr \
  proftypexl.ovr \
  slprofxl.ovr \
  strintxl.ovr
do
  orgFile=<path to the data files>/${PatchFile}
  destFile=<path to the domain>/Curve/input/${PatchFile}

  if [[ -f "${orgFile}" ]]; then
    cp -f "${orgFile}" "${destFile}"
  fi
done
```

4. Enter the following command to build a domain for the Curve configuration:

```
rpasInstall -fullinstall -dh <path to the domain> -cn Curve -ch <path
to the configuration> -in <path to the data files> -log <path to the
location and name of the installation log> -rf AppFunctions –rf
ClusterEngine –p pgrp
```

5. After the domain installation has completed the sales data must be loaded into the domain using the `loadmeasure` utility. Open a command prompt from the master domain (/Curve) and type the following commands:

   ```
   loadmeasure -d . -measure dpos
   ```

   ```
   loadmeasure -d . -measure rsal
   ```

   ```
   loadmeasure -d . -measure csal
   ```

   ```
   loadmeasure -d . -measure psal
   ```

6. Open a command prompt from the local domain (/Curve/ldom0) and type the following command:

   ```
   mace -d . -run -group common_batch
   ```

   Repeat this step for each of the remaining local domains (/Curve/ldom1, /Curve/ldom2).

Start the RPAS Server (DomainDaemon)

In order to use the domains built from the sample configurations, the RPAS Server must be running on the server/machine where the domain is located.

The RPAS Server is started by running the RPAS DomainDaemon executable, which provides a centralized process for managing domain connections between the client and the server.

Following is an example of running the DomainDaemon, which allows you to connect to the RPAS Server and a domain using the RPAS Clients. When running the DomainDaemon, RPAS requires the SSL settings to be defined. Different settings are required depending on the client being used. This example is a simple case of using the Classic Client with no-SSL. Complete information about the Domain Daemon and the different SSL options is located in the Classic Client and Fusion Client versions of the *Oracle Retail Predictive Application Server Administration Guide*.

Run the following command from a UNIX command line (or using Cygwin on Windows). If the environment variables paths have been properly set, this command can be run from any directory.

Example 4–3 Running the DomainDaemon

```
DomainDaemon -port <port_number> -start –ssl <option> [-wallet <path_to_wallet>]
```

Where `<port_number>` is an integer between 1025 and 65535.

Where `<option>` is the type of SSL you are using.

Where `<path_to_wallet>` is the path to the Oracle Wallet containing the server side SSL certificates. This is optional depending on which SSL option you are using.

This port number is used when configuring the client connection. Refer to the "Installing and Configuring the RPAS Classic Client" and "Installing the RPAS Fusion Client" chapters for additional information.

For additional information about setting up SSL wallets, refer to Setup SSL

SSL Options

DomainDaemon lets you set up SSL with these options:

Table 4–1 SSL Option Values and Recommendations

Value	Description	Application	Usage	Recommendation
1	One-way SSL with server authentication.	Classic Client	One-way authentication and encrypted data transfer.	Most secure -recommended.
2	Two-way SSL with client and server authentication.	Fusion Client	Two-way authentication and encrypted data transfer.	Most secure -recommended.
3	SSL without authentication.	Classic Client	Encrypted data transfer without authentication.	Vulnerable to man in the middle attacks - not recommended.
4	Two-way SSL authentication only.	Fusion Client	Two-way authentication but unencrypted date transfer.	Least secure - not recommended except for users that deploy everything behind a firewall.

5

Installing the RPAS Fusion Client

The RPAS Fusion Client is the Web-based Rich Client for the Retail Predictive Application Server (RPAS) platform developed using the latest Oracle Application Development Framework (ADF).

This chapter describes typical installations of the RPAS Fusion Client. It includes these sections:

- Road Map for Installing the RPAS Fusion Client
- Pre-Installation Tasks
- Installation Tasks
- Postinstallation Tasks
- Troubleshooting

> **Note:** The RPAS Fusion Client is different from the RPAS Web Deployment. For more information, refer to Chapter 7, "RPAS Classic Client Web Deployment".

Road Map for Installing the RPAS Fusion Client

This section explains how to install and set up the RPAS Fusion Client application, along with the required and optional software.

These instructions assume knowledge of application servers, databases, and application installation or administration, and are intended for system administrators and experienced IT personnel. Before carrying out any of these activities, ensure that you understand UNIX commands (including shell configuration and scripting), directory operations, and symbolic links.

In order to install RPAS Fusion Client for production, you must perform the following tasks listed in Road Map Phases and Tasks.

Road Map Tasks

To install the RPAS Fusion Client, there are three phases and tasks within each phase as listed in Table 5–1.

Table 5–1 Road Map Phases and Tasks

Phase	Task	Additional Information
Pre-Installation Tasks	Plan your environment, based on your business needs.	For more information on the planning process and the supported configurations, refer to Chapter 2, "Getting Started" and the section, Planning.
	Install and set up the RPAS Infrastructure.	For more information, refer to Chapter 3, "Installing on UNIX and Linux Environments" or Chapter 4, "Installing on a Windows Environment".
	Decision for OBIEE reports from Fusion Client	If using the OBIEE reports feature, then also select the **Oracle Enterprise Manager - 12.1.3.0 [em]** check box during the WLS/ADF 12c Setup.
	Install Oracle database for WLS/ADF 12c setup	For more information, refer to Install Oracle Database for WLS/ADF 12c Setup
	Set up the WebLogic server.	For more information, refer to Setting Up the WebLogic Server.
	Access the RPAS Fusion Client installation software.	For more information, refer to Accessing the Fusion Client Installation Media.
	Set up the `install.properties` file.	For more information, refer to Setting Up Your Installation Properties File.
	Set the environment variables.	For more information, refer to Setting Up Environment Variables.
	Optional: Validate the WebLogic and ADF versions.	For more information, refer to Validating WebLogic and Oracle ADF Versions.
	Create credentials in the Oracle Wallet.	For more information, refer to Creating User Credentials in an Oracle Wallet.
Installation Tasks	Install Solution Plug-ins	For more information, refer to Appendix: Installing Solution Plug-ins or the installation instructions for the specific Oracle Retail application.
	Install the Fusion Client in silent mode or text/graphical mode.	For more information, refer to Installing the RPAS Fusion Client in Silent Mode or Installing the RPAS Fusion Client in Graphical or Text Mode.
Postinstallation Tasks	Clear the browser cache.	For more information, refer to Clear the Browser Cache.
	Configuring External Authentication.	For more information, refer to Configuring External Authentication.
	Setup SSL between the RPAS server and the RPAS Fusion Client.	For more information, refer to Setup SSL.
	Review the RPAS configuration property files.	This task is only necessary when installing the RPAS Fusion Client over an existing RPAS Fusion Client environment. For more information, refer to Review the RPAS Configuration Property Files.
	Optional: Set up SSO.	For more information, refer to Set Up Single Sign-On (SSO).
	Optional: Set up the configuration properties file, pivot table styles, and layout and formatting.	For more information, refer to the *Oracle Retail Predictive Application Server Administration Guide for the Fusion Client*.
	Install and set up the RPAS solution to work with the RPAS Fusion Client.	For more information, refer to the Installation Guide of the relevant RPAS solution.

Pre-Installation Tasks

The pre-installation process includes these tasks that must be performed in order:

Table 5–2 Order of Pre-Installation Tasks

Order	Task
1.	Planning
2.	Install and Set Up the RPAS Infrastructure
3.	Install Oracle Database for WLS/ADF 12c Setup
4.	Setting Up the WebLogic Server
5.	Setting Up the Domain Policy Store. Optional step for use with an OID-based policy store.
6.	Accessing the Fusion Client Installation Media
7.	Setting Up Your Installation Properties File
8.	Setting Up Environment Variables
9.	Validating WebLogic and Oracle ADF Versions
10.	Creating User Credentials in an Oracle Wallet

Planning

This is the first pre-installation task.

Before installing the RPAS Fusion Client, you must first determine the performance and availability goals for your business, and then plan the hardware, network, and storage requirements accordingly. This section provides some basic considerations for the installation. It also includes the list of hardware and software requirements.

This section includes the following topics:

- Planning Your Environment
- Supported Configurations

Planning Your Environment

Planning your implementation prior to an installation also gives you a better understanding of the environment, and enables you to adapt faster to any future changes in the environment setup.

Use the following steps to plan and prepare the product environment:

1. Plan and design the infrastructure, based on your business needs, for the installation. This includes:
   - Meeting the hardware and associated software requirements.
   - Acquiring the prerequisite software (and licensing).
   - Determining the external authorization strategy. For more information, refer to Configuring External Authentication.
   - Determine if you are installing any solution plug-in. For more information, refer to Appendix: Installing Solution Plug-ins.
   - Setting up the load balancers and clusters. For more information, refer to Considerations for Setting Up Load Balancers.
   - Gathering the capacity data.

- Planning the data security policies.
- Designing the backup and recovery strategies.

2. Determine the size of the implementation.
3. Identify source systems. Identify the systems that will exchange data with RPAS Fusion Client.

Considerations for Setting Up Load Balancers

You can choose to implement a software load balancer or network-based load balancer hardware.

Before you start setting up a load balancer, you must consider the items listed in Table 5–3. These considerations are relevant for an external load balancer used in a multiple managed application server deployment featuring standalone authentication. If using Oracle SSO 11g, the `mod_wl_ohs` Web Tier plug-in performs the application server level load balancing function.

An external load balancer can still be beneficially used in an SSO deployment. This would be deployed in front of multiple web tier servers. The advantage is in avoiding single point of failure (at the web server), and optionally, providing Secure Sockets Layer (SSL) termination, compression, and static content caching at the load balancer.

Table 5–3 External Load Balancer Considerations - Non-SSO Deployment

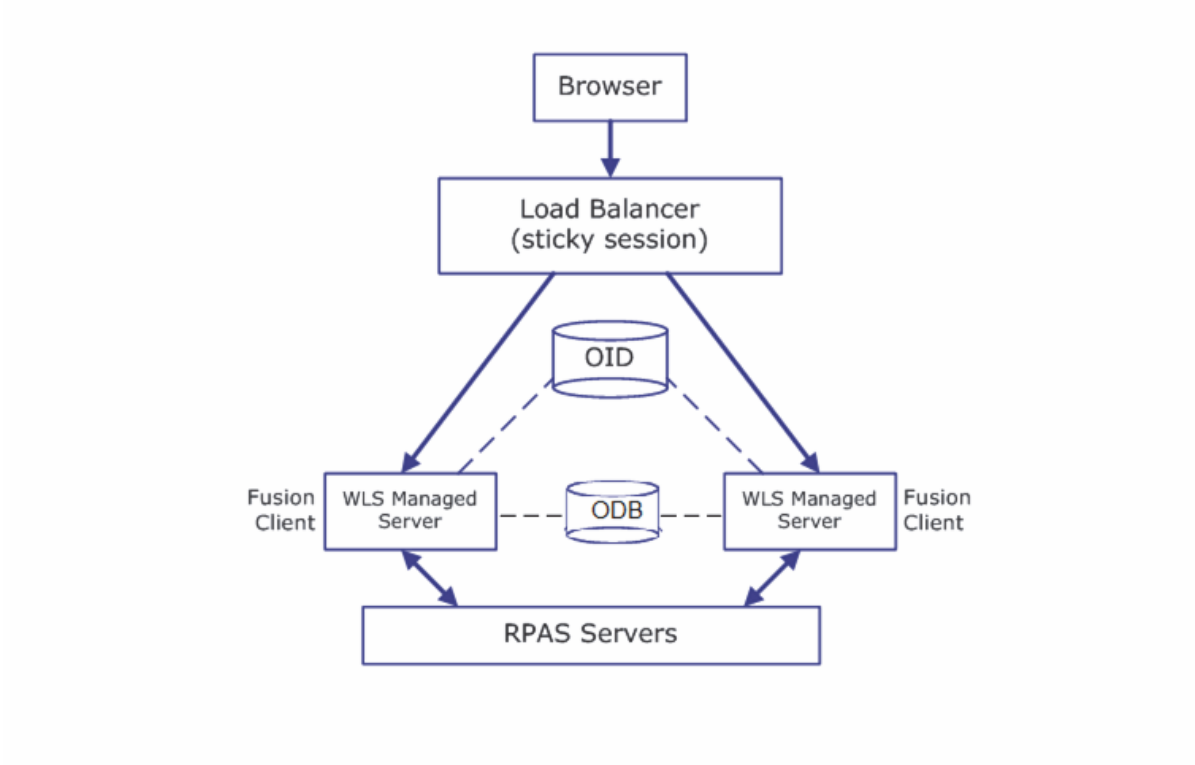
Item	Description
SSL Termination at the load balancer	This establishes a Secure Sockets Layer (SSL) protocol at the load balancer and replaces the need for the Web server to set up the SSL. To set up SSL Termination at the load balancer, ensure that it is configured with an SSL certificate (self generated or signed by a certificate authority).
Load Balancing Method	It is recommended to use a Round Robin load balancing method, coupled with session affinity. In the Round Robin method, requests are balanced across a list of available servers and servers are selected sequentially. By coupling with session affinity, subsequent requests from a specific user are redirected to the same server assigned for the previous requests from the user. Session affinity is necessary because there is no application state replication between the servers. This will avoid the excessive need for the application state to be replicated between the servers.
KeepAlive	The load balancer uses the KeepAlive feature to test the servers before directing the users to an active server. This test typically involves setting up a keepalive page (such as `index.html`) or a custom page that the load balancer will attempt to load as per the test. You can choose to set up this feature or set up a monitor that checks the ports on the servers ensuring that they are active. Setting up a monitor is the preferred method.

Non-SSO deployment

The considerations listed in Table 5–3 apply in case of non-SSO deployment. Figure 5–1 shows a load balancer for a non-SSO deployment.

Using a load balancer is recommended for scenarios where you need to use multiple servers because one server may not be able to handle the load. The RPAS Fusion Client can be installed and used without implementing a load balancer.

Figure 5–1 Load Balancer for a Non-SSO Deployment

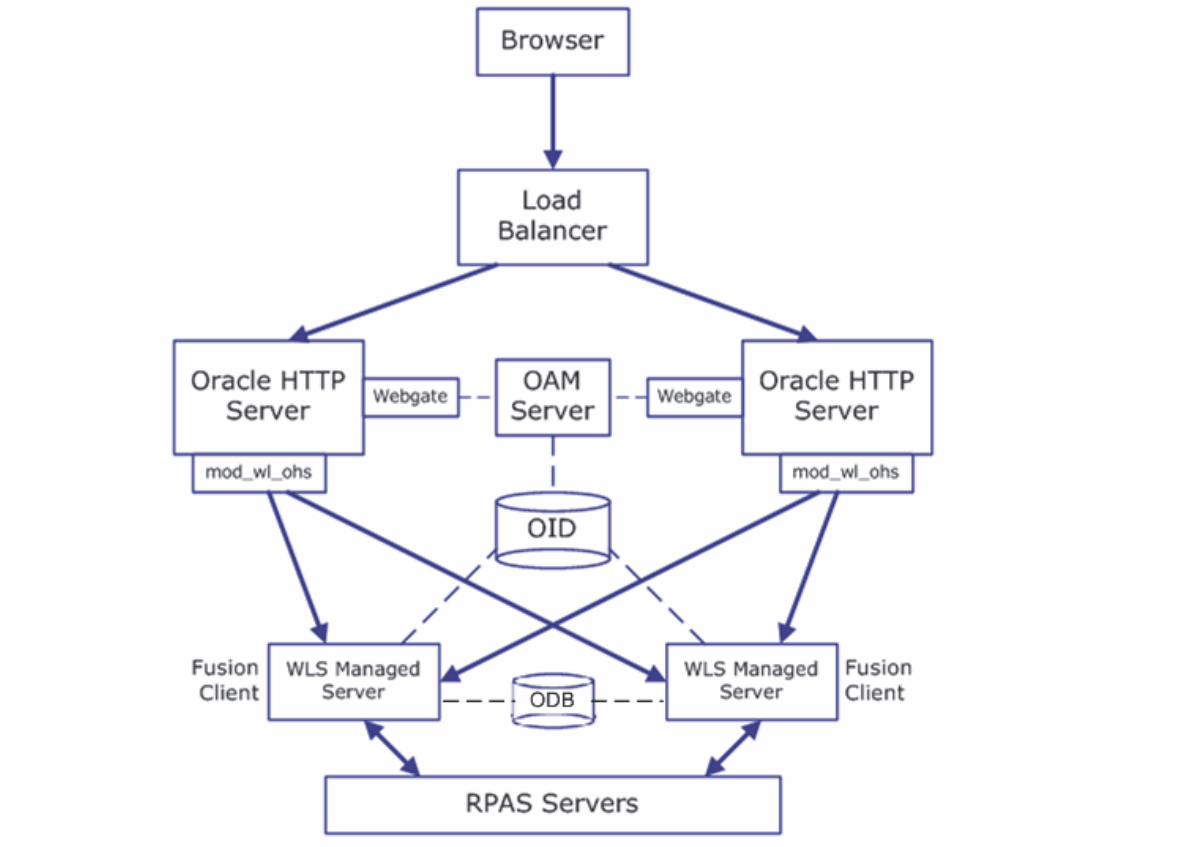


SSO deployment

These considerations apply in case of an SSO deployment. Figure 5–2 shows a load balancer for an SSO deployment.

In a deployment that uses Oracle Single Sign On for authentication, you must use the `mod_wl_ohs` Oracle HTTP server module to distribute requests with session affinity across multiple WebLogic managed servers. Single Sign On is enforced by an Oracle Access Manager 11g WebGate module plugged into the Oracle HTTP server. Scalability on the application server is achieved through the use of multiple managed servers. Single point of failure can be avoided by deploying multiple Oracle HTTP servers with a load balancer in front. The load balancer can also bring in other performance benefits such as SSL termination, content compression, caching, and so on.

Figure 5–2 Load Balancer for an SSO Deployment



For more information on setting up load balancers, refer to the documentation included with the Load Balancer.

Session Cookie Name

There is an important consideration when using a Web Tier with a load balancer - such as the mod_wl_ohs Web Tier plug-in - to mediate requests to multiple WebLogic managed servers. Load balancers often implement session affinity (also called sticky sessions) using the WebLogic Server session cookie. The session cookie has been renamed from the default. The new name is `RPASJSESSIONID`.

If using mod_wl_ohs as the load balancing Web tier plug-in, then add the following directive to the plug-in configuration:

```
WLCookieName RPASJSESSIONID
```

Example 5–1 Session Cookie

This example is for two managed servers with the context root as: rpas_sso.

```
<Location /rpas_sso>
  SetHandler weblogic-handler
  WebLogicCluster burrg41101v:8421,burrg41101v:8431
  WLCookieName RPASJSESSIONID
 </Location>
```

Supported Configurations

For more information on the software and hardware requirements, refer to Table 1–3, " RPAS Fusion Client Hardware and Software Requirements" in Chapter 1, "Introduction".

Note: RPAS Fusion Client is included with the RPAS installation media and requires that the RPAS Server and Configuration Tools are upgraded to Release 14.1.2.

Install and Set Up the RPAS Infrastructure

This is the second pre-installation task, ensure that all previous pre-installation tasks are complete.

Install and set up the RPAS Infrastructure. For more information, refer to Chapter 3, "Installing on UNIX and Linux Environments" or Chapter 4, "Installing on a Windows Environment".

Install Oracle Database for WLS/ADF 12c Setup

This is the third pre-installation task, ensure that all previous pre-installation tasks are complete.

Perform the following steps to set up WebLogic 12c with ADF Runtime:

1. Install the Oracle 12c Database. Download the 12c Database from the Oracle Software Delivery Cloud: `https://edelivery.oracle.com/`

   Note: AL32UTF8 should be selected as the option while installing it. Refer to Weblogic/ADF12.1.3 supported platforms documentation for other certified database versions.

2. Download and Install of ADF Runtime 12.1.3. Download ADF Runtime 12.1.3 from OTN. This package is bundled with WebLogic 12c. For more information, see the chapter, Deploying ADF Applications in *Oracle Fusion Middleware, Administering Oracle ADF Applications, 12c (12.1.3)*.

   Note: The WebLogic Server 12c (Release 12.1.3) and ADF use Oracle Platform Security Services (OPSS); use by the Oracle 12c Database is not required of the Fusion Client application.

Setting Up the WebLogic Server

This is the fourth pre-installation task, ensure that all previous pre-installation tasks are complete.

Note: ADF 12.1.3 is installed as a part of the WebLogic server 12.1.3 — you must apply the ADF patches separately.

The RPAS Fusion Client is a Web-based client for RPAS. When you run the Fusion Client installer, the installer will require a domain set up over the WebLogic Server to deploy the Fusion Client as an application.

Before installing the RPAS Fusion Client, you must install the WebLogic Server and set up a domain for the Fusion Client. This chapter describes how you can set up the WebLogic Server for the Fusion Client. It includes the following sections:

- Installing the WebLogic Server
- Applying the Oracle ADF Run Time Patches
- Running the RCU
- Creating Schemas
- Setting Up a WebLogic Domain

Note: Once you set up the WebLogic server and domain, you must take note of the location where you installed the WebLogic domain. You will need to set up this location as an environment variable, WEBLOGIC_DOMAIN_HOME, before running the Fusion Client installer.

Installing the WebLogic Server

Install the Oracle WebLogic Server and Application Development Runtime. For WebLogic Server installation instructions, refer to the Oracle WebLogic Server Documentation for guidance. For Application Development Runtime installation, refer to the *Oracle Fusion Middleware Installation Guide for Application Developer*.

In the following sections, the WebLogic installation directory is referred to as the **<MW_HOME>** directory.

Applying the Oracle ADF Run Time Patches

> **Note:** ADF12c is installed bundled with WebLogic 12c. There is no need to separately install it.

You must apply the ADF Run Time Patches 18886249, 19473060, and 21832568 on top of
ADF 12c (12.1.3).

> **Note:** When installing multiple versions of WebLogic on a system, it is recommended that you modify the base middleware directory to include a reference to the WebLogic Server version.

To download and apply the patches:

1. Log on to the My Oracle Support Web site and download the patches 18886249, 19473060, and 21832568.

   > **Note:** After completion of the first patch installation, all steps should be repeated to download and apply the second patch.

   To download these patches:

   a. In a Web browser, open the My Oracle Support Web at this URL:

      `https://support.oracle.com/`

   b. Select a language and sign on to the Web site by clicking **Sign In**.

   c. Once signed in, the **My Oracle Support | Dashboard** window opens.

   d. Click the **Patches & Updates** tab.

   e. On the **Patch & Updates** window, under **Patch Search**, click **Patch ID** or **Number**.

   f. In the **Patch ID** or **Number** field, enter *18886249*.

   g. Optionally, you can also choose a platform from the Platform drop-down list.

   h. Click **Search**. The **Patch Search Results** window opens.

   i. In the **Patch Search Results** window, under **Patch ID**, click the relevant patch.

   j. On the next window, click **Download**. It is located on the left side of the window.

> **Note:** On the **Patch Search Result**s window, you can also select the row that matches the patch description, and then click **Download** on the toolbar that appears under the selected row.

2. Unpack the ZIP file to a temporary directory and navigate to this location.
3. Set the ORACLE_HOME and PATH environment variables using the following commands:

   ```
   export $ORACLE_HOME=$MW_HOME/oracle_common
   ```

   ```
   export PATH=$PATH:$ORACLE_HOME/OPatch
   ```

4. At the command prompt, run the following command to apply the patch:

   ```
   opatch apply
   ```

5. Follow the prompts to complete the patch installation. For detailed instructions, refer to the `README.txt` file included in the patch directory.

> **Note:** Repeat Steps 1 - 5 to download and apply the 19473060 and 21832568 patches entering *19473060* and *21832568* in Step 1f.

You can now set up your WebLogic domain. For more information, refer to Setting Up a WebLogic Domain.

Running the RCU

Run the Repository Creation Utility (RCU).

> **Note:** RCU 12.1.3 comes with the Oracle Fusion Middleware Infrastructure (Not as separate software), $ORACLE_HOME/oracle_common/bin).
>
> For more information, see *Oracle Fusion Middleware, Administering Oracle ADF Applications, 12c (12.1.3)*.

The following items must be set when installing RCU:

1. Set ORACLE_HOME= DATABASE_INSTALL_DIR
2. Change the configuration of Database to opena pluggable Database

   Login as sysdba--> sqlplus / as sysdba

   alter pluggable database all open;

   exit from SQL Prompt

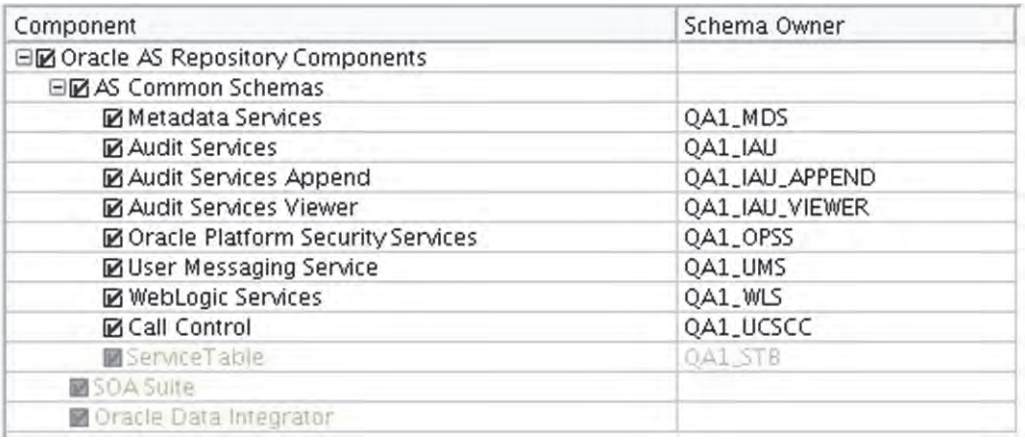
   Do LSNRCTL> status to find the name of the Pluggable Database. For example, `pdborcl.idc.oracle.com`
3. Set ORACLE_HOME= ADF_RUNTIME_INSTALL_DIR/oracle_common
4. Go to ORACLE_HOME/bin using Command Prompt and set RCU.bat as interactive
5. Give SID as the pdborcl.idc.oracle.com during INSTALL

Creating Schemas

Create schemas. The following schemas must be created (where XXXX is a prefix):

- XXXX_MDS
- XXXX_IAU
- XXXX_IAU_APPEND
- XXXX_IAU_VIEWER
- XXXX_OPSS
- XXXX_UMS
- XXXX_WLS
- XXXX_UCSCC

Figure 5–3 12c Schemas



Setting Up a WebLogic Domain

Use the WebLogic Configuration Wizard to create and set up a domain on the WebLogic Server. This section describes how you can create and set up a domain. It also introduces the steps to configure the managed servers and clusters on the application server. For more information on the WebLogic Configuration Wizard and customizing the domain environments with managed servers, refer to the Oracle Fusion Middleware Creating Domains Using the Configuration Wizard.

> **Note:** For headless installations, ensure that you set up the WebLogic Startup script with the java.awt.headless parameter. For more information, refer to Troubleshooting.

To set up a WebLogic domain:

1. Navigate to the **<MW_HOME>/common/bin** directory, and run the following command to start the WebLogic Configuration Wizard in the graphical mode:

   ```
   sh config.sh
   ```

2. On the WebLogic Configuration Wizard, follow the steps listed in Table 5–4:

Table 5–4 Steps to Setup a WebLogic Domain

Step	Window	Task
1.	Welcome Window	Click the Create a new WebLogic domain option, and then click Next.
2.	Select Domain Source Window	Click the Generate a domain configured automatically to support the following products option, select the Oracle JRF - 12.1.3.0 [oracle_common] check box, and click **Next**. Notes: ■ If the JRF option is not available, you must exit the installation, install the Application Development Runtime, and then restart the domain configuration. ■ The **Basic WebLogic Server Domain - 12.1.3.0** check box is automatically selected and unavailable. ■ If using the OBIEE reports feature, then also select the **Oracle Enterprise Manager - 12.1.3.0 [em]** check box.
3.	Specify Domain Name and Location Window	Enter a domain name in the Domain Name field. In the **Domain location** field, specify the location where you want to install the domain. This location is referred to as the WEBLOGIC_DOMAIN_HOME all through this document.
4.	Configure Administrator User Name and Password Window	Set up an administrative user name and password. Important: Keep a note of the user name and password. You must specify this user name and password in the `ant.install.properties` file. The Oracle Installer uses this user to connect to the WebLogic Server during the RPAS Fusion Client installation.
5.	Configure Server Start Mode and JDK Window	Under WebLogic Domain Startup Mode, click Production Mode. Under JDK Selection, select the relevant JDK. Click **Next**.
6.	Database Configuration	Select the RCU Data default option. DBMS/Service: The database service ID (for example, orcl). Host name: The DNS name of the database machine. Port: the TNS listener port (for example, 1521). Schema Owner: The name of the service schema created by RCU. (for example ,DEV_STB) Schema Password: The password for the service schema.
7.	Select Optional Configuration Window	Select the configurations you want to customize and click **Next**. Go to the Configure the Administration Server Window step or proceed directly to creating your domain by skipping the following steps and going to the Configuration Summary Window step.
8.	Configure the Administration Server Window	Enter relevant information in the following fields: ■ Name - Valid server name. (String of characters that can include spaces.) ■ Listen address - Listen address for a server instance. ■ Listen port - Valid value for the listen port. ■ SSL listen port - Valid value to be used for secure requests. ■ SSL enabled - Select this check box to enable SSL. You can enter values in the SSL listen port field once you select this check box. Click **Next**.

Table 5–4 (Cont.) Steps to Setup a WebLogic Domain

Step	Window	Task
9.	Configure Managed Servers Window	**Note:** OPSS services need to deployed on all servers. Click **Add**, and then enter relevant information in the following fields: ■ Name - Valid server name. (String of characters that can include spaces.) ■ Listen address - Listen address for a server instance. ■ Listen port - Valid value for the listen port. ■ SSL listen port - Valid value to be used for secure requests. Repeat this step to add more managed servers. Click **Next**.
10.	Configure Clusters Window	This window opens, once you specify the managed servers. Click **Add**, and then enter relevant information in the following fields: ■ Name - Valid cluster name. (String of characters that can include spaces.) ■ Multicast address - Address used by the cluster members to communicate with each other. ■ Multicast port - Port used by the cluster members to communicate with each other. ■ Cluster address - Address that identifies the Managed Servers in the cluster. Repeat this step to specify more clusters. Click **Next**.
11.	Assign Servers to Clusters Window	Use the arrow buttons and assign the servers to the clusters specified in the domain. Click **Next**.
12.	Configure Machines Window	Click **Add**, and then add the machine (UNIX-based) information where the Fusion Client will be deployed. If you need to specify a local host because one of your target managed server is on the local host (meaning that it is the same as your admin server host), enter any of the following: ■ localhost ■ 127.0.0.1 ■ the DNS name Do not enter an IP address of the local host because it will try to use SSH to copy files to the local host. This is not only unnecessary but may not work depending on the SSH environment and on whether you specified SSH properties. In a clustered installation involving multiple machines, this includes all the systems where the RPAS Fusion Client will be deployed. In other words, the installer looks for all machines defined in this section in order to copy files to each of them. If you do not want to install the product on a host, do not specify that host here. **Note:** Only define machines where you plan to run managed servers. Any additional machine definitions may unnecessarily cause the installer to copy configuration files to that machine. Click **Next**.
13.	Assign Servers to Machines Window	Use the arrow buttons and assign the managed servers to the machines specified in the domain. Click **Next**.

Table 5–4 (Cont.) Steps to Setup a WebLogic Domain

Step	Window	Task
14.	Target Deployments to Clusters or Servers	In the left pane, select the clusters or servers, and then select the relevant application check boxes in the right pane to target them to the specific cluster or managed server. For each cluster and managed server, select the **Library** check box. The WebLogic domain must be set up in such a manner that all the clusters and the relevant managed servers include all the libraries included with the WebLogic server.
15.	Target Services to Clusters or Servers	In the left pane, select the clusters or servers, and then select the relevant services check boxes in the right pane to target them to the specific cluster or managed server.
16.	Configuration Summary Window	Review and confirm the configuration summary. Click **Next**.
17.	Creating Domain Window	Displays the domain configuration progress. After the configuration is complete, click **Done**.

Setting Up the Maximum Heap Size

After you have set up the WebLogic domain, ensure that you set up the maximum heap size for the WebLogic server. Setting a maximum heap size depends on your implementation.

For more information on heap sizing, refer to the Oracle Java documentation on Java Performance Tuning and the *Oracle Fusion Middleware Performance and Tuning for Oracle WebLogic Server*.

Setting Up the Garbage Collection Parameter

Setting up the garbage collection parameter for WebLogic server provides better performance with Fusion Client. Refer to the following links for more details.

Note: For information about setting Thread Counts, refer to:

`http://docs.oracle.com/cd/E23943_01/web.1111/e13814/toc.htm`

For IBM/Java

-Xgcpolicy:gencon

Requests the combined use of concurrent and generational GC to help minimize the time that is spent in any garbage collection pause.

`http://publib.boulder.ibm.com/infocenter/javasdk/v6r0/topic/com.ibm.java.doc.user.aix64.60/user/garbage.html`

For Oracle/Hotspot

-XX:+UseConcMarkSweepGC

`http://www.oracle.com/technetwork/java/javase/gc-tuning-6-140523.html`

For Linux

On a Linux headless server, you should implement the two options covered in Troubleshooting if your installation hardware configuration requires them. This prevents problems when deploying or using the Fusion Client.

Note: This does not fall under the Garbage Collection parameter.

Configuring the Node Manager for Clusters-based Installation

When setting up a WebLogic cluster with multiple physical machines, ensure that you have completed the following tasks:

- Installed WebLogic Server and the ADF Runtime libraries on identical paths on each machine. For example, the identical path could be:

  `/u00/webadmin/Oracle/Middleware`

  This path must exist on each machine's filesystem. If this is not done, the essential ADF runtime libraries will not deploy and the application will not run.

- Copy the WebLogic domain structure from the Admin Server machine to all the managed server machines, ensuring identical filesystem locations. You can use the `pack.sh` and `unpack.sh` utilities to perform the copy. This process also copies the `startWebLogic.sh` script, which can be found in the domain directory. This script can be used to start up the managed servers.

Note: The previously listed considerations do not apply if the WebLogic Server and the domain are installed on a shared filesystem.

There are two startup options for customizing managed server startup parameters: manual startup and Node Manager startup. These options are described in the following sections.

Manual Startup

The managed servers in a WebLogic cluster can be started manually by using the `startManagedWebLogic.sh` script in the WebLogic domain directory. You can modify this script to customize server startup parameters such as JVM heap size, garbage collection settings, and so on.

Node Manager-Based Startup

Using NodeManager, choose one of the following options for server startup customization:

- Setup custom server startup parameters for each managed server on the WebLogic Administration Console. These parameters can be accessed on the Server Start tab under **Home -> Servers -> <managed-server-name>**.
- Specify startup parameters in the `startup.properties` file located in the following filesystem path:

  <domain-home>/servers/<managed-server-name>/data/nodemanager

  This file must exist on each server machine's filesystem.

Note: If a shared filesystem is in use, all the `startup.properties` files must reside on that filesystem.

- Make NodeManager use a start script. The script on each server can be modified as needed, as described in Manual Startup. Use `startWebLogic.sh` as the base script and then modified as needed.

For more information about cluster-based installation, including information on installing a WebLogic server and setting up WebLogic clusters, refer to the *Oracle Fusion Middleware Node Manager Administrator's Guide for Oracle WebLogic Server*.

Setting Up the Domain Policy Store

This is the fifth pre-installation task, ensure that all previous pre-installation tasks are complete.

Note: This step is optional and is for use with an OID-based policy store.

By default, a file-based policy store is created when a WebLogic domain is created. It is supported, but be sure to set up controls to prevent unauthorized access to the policy files (located inside the domain directory).

An LDAP policy store is also supported by the Fusion Client. Currently only Oracle Internet Directory (OID) is supported. Refer to Upload Application Security Policies to OID-based Domain Policy Store for more information.

Note: DB-based policy store is not supported at this time.

For guidance on setting up an OID-based policy store, refer to the chapter "Configuring the OPSS Security Store" in the Oracle® Fusion Middleware Application Security Guide 11g Release 1 (11.1.1) Configuring the OPSS Security Store. This document is available through My Oracle Support.

Accessing the Fusion Client Installation Media

This is the sixth pre-installation task, ensure that all previous pre-installation tasks are complete.

The RPAS Fusion Client installation media is included with the RPAS installation media. The installation files for the RPAS Fusion Client are available at the following location in the **<RPAS_CD_IMAGE>** directory:

```
<RPAS_CD_IMAGE>/FusionClient.zip
```

Note: The directory, **<RPAS_CD_IMAGE>**, is the temporary location where the RPAS installation media files were unpacked.

Setting Up Your Installation Properties File

This is the seventh pre-installation task, ensure that all previous pre-installation tasks are complete.

Note: It is recommended that you run the graphical installer the first time you are installing the RPAS Fusion Client. The installer creates a installer properties file for you.Refer to Installing the RPAS Fusion Client in Graphical or Text Mode

Later on if you need to reinstall the RPAS Fusion Client, you can use this properties file (making any changes you see fit) and run the installer in silent mode. Refer to Installing the RPAS Fusion Client in Silent Mode

To install the RPAS Fusion Client, it is recommended that you set up the installation properties file (`ant.install.properties`) before running the installer.

Note: For an installation in silent mode, you must set up the installation properties file before running the installer. For an installation in graphical or text mode, this is optional.

To set up your `ant.install.properties` file:

1. Navigate to the RPAS Fusion Client directory, copy the `ant.install.properties.template` file to the same directory, and rename it `ant.install.properties`.
2. Edit the `ant.install.properties` file using any text editor, specifying values as described within the file, and save it. For more information on the parameters, refer to Installation Properties File Parameter Reference.

Note: Ensure that the `ant.install.properties` file is available in the same directory with the `install.sh` script.

Installation Use Cases

The following use cases can help you understand what to specify for various installation properties which have values that depend on how your domain is configured and whether you are installing for the first time. Your installation may be similar to one or multiple use cases. If your installation is a combination of use cases, all properties from the relevant use cases apply.

Installation Use Case 1

Installation on a Single Managed Server, Same Host as Admin Server

Installation on a single managed server located on the same physical host as the admin server:

- Set `input.appserver.host` to either:
  - `localhost`
  - `127.0.0.1`.
- Since the managed server is on the same physical host as the admin server, set `input.is.multiple.hosts` to **No**.
- Any property that begins with `input.ssh` (SSH-related properties) is inconsequential and can be set to either **Yes** or **No**.

Installation Use Case 2

Installation on the Admin Server

Use the same guidelines as described in Installation Use Case 1.

Installation Use Case 3

Installation on a Single Managed Server, Different Host from Admin Server

Installation on a single managed server that is located on a different physical host as the admin server host:

- Set `input.is.multiple.hosts` to **Yes**.
- If using silent install, then you must set `input.retrieve.credentials` to **Yes** and prepare the wallet with WebLogic and SSH credentials, before install. Set it to **No** if you do not have SSH credentials stored in the wallet and need to specify the credentials yourself.

> **Note:** Setting `input.retrieve.credentials` to **Yes** implies that you have also saved the WebLogic admin credentials in the wallet. Either all user credentials are to be retrieved from the wallet (for WebLogic and SSH) or they are to be entered manually by the user in the installer UI.
>
> If using the silent mode installer, then you must set `input.retrieve.credentials` to **Yes** and prepare the Wallet with WebLogic and SSH credentials, before installing.

All other SSH-related properties that begin with input.ssh must be set or ignored appropriately, so that files can be copied to the managed server host from the current host which is the admin server host.

> **Note:** It is required that all machines have the same SSH UserID and password.
>
> Because the installer uses SCP to migrate files to remote hosts, you should ensure that you can copy files from the installation host to the target server using SCP and the specified parameters. If you cannot, the installer will fail.

Installation Use Case 4

Installation on a Cluster of Managed Servers, Same Host as Admin Server

Installation of a cluster of managed servers, where all managed servers are located on the same physical host as the admin server:

Use the same guidelines as described in Installation Use Case 1.

Installation Use Case 5

Installation on a Cluster of Managed Servers, Different Host as Admin Server

Installation onto a cluster of managed servers, where at least one of the managed servers is located on a different physical host as the admin server host:

Use the same guidelines as described in Installation Use Case 3

Installation Use Case 6

Installation on Multiple Standalone Managed Servers, Same Host as Admin Server

Installation on multiple standalone managed servers that are not part of a cluster, where all managed servers are on the same physical host as the admin server host:

Use the same guidelines as described in Installation Use Case 1

Installation Use Case 7

Installation on Multiple Standalone Managed Servers, Different Host as Admin Server

Installation on multiple standalone managed servers that are not part of a cluster, where at least one managed server is located on a different physical host as the admin server host:

Use the same guidelines as described in Installation Use Case 3.

Installation Properties File Parameter Reference

Table 5–5 describes the parameters in the `ant.install.properties` file that you must set up before you install the RPAS Fusion Client application:

Note: When the installation starts, values set for all the user credentials will be encrypted and stored in the Oracle Wallet, and then cleared from the `ant.install.properties` file. Therefore, you will have to change the installation parameters to retrieve the credentials before attempting to re-install.

Table 5–5 Installation Properties File Parameter Reference

Parameter Name	Description
Retrieve Credentials	
input.retrieve.credentials	As mentioned previously, this applies to GUI mode of install. For silent, you must set it to **Yes**. You should set the value to **Yes** if you have stored user credentials to retrieve them later. If you do not have user credentials stored in the secure wallet, or if you want to overwrite the existing credentials or store new credentials, set the value to **No**. If using the silent mode installer, then you must set `input.retrieve.credentials` to **Yes** and prepare the Wallet with WebLogic and SSH credentials, before installing.
Target Installation Directory	
input.install.target.dir	Specify the location where you want to install the RPAS Fusion Client.
Logs and Temporary Directories	
input.wallet.dir	Specify the location of the Oracle Wallet where the user credentials are to be saved or retrieved. This cannot be in or under the same temporary directory indicated by input.install.tmp.dir.
input.app.log.dir	Specify the location for the application log files.

Table 5–5 (Cont.) Installation Properties File Parameter Reference

Parameter Name	Description
input.install.log.dir	Specify the location for the installation log files.
input.install.tmp.dir	Specify the location for the temporary file directory used during installation.
WebLogic Admin Server Information	
input.appserver.host	Specify the host name where the application server is running. If the application server is running on the same host as the installer. If you need to specify a local host because one of your target managed server is on the local host (meaning that it is the same as your admin server host), enter any of the following: ■ localhost ■ 127.0.0.1 ■ the DNS name If the application server is running on a different host as the installer, enter the DNS name or IP address of the host where the application server is running. You must also specify values for appropriate parameters starting with input.ssh.
input.admin.server.port	Specify the port number associated with the application server.
input.admin.username	Specify the administrative user name for the application server.
input.admin.username.alias	Specify an alias name for the administrative user. Specifying an alias name for the administrative user enhances the security for the application. When left blank, the alias name will default to the administrative user name.
Application Configuration Information	
input.is.multiple.hosts	To specify that your domain contains at least one managed server machine that is different from the admin server machine, set the value to yes. If all the managed server machines are on the same machine as the admin server machine, then set the value to no. In other words, even if you have a cluster with multiple managed servers or have multiple standalone managed servers, the value should be no if all of these managed servers are on the same machine as the admin server machine.
input.sso.enabled	Specify whether you want to install the application to be SSO enabled (set the value to yes). To install the application without configuring SSO, set the value to no. For more information on setting up SSO, refer to the *Oracle Retail Application Server Administration Guide for the Fusion Client*.
SSH Credentials	Applies to cluster-based installations only.
input.ssh.authentication.mode	Specify one of the following authentication methods: ■ password - Use the specified password (associated with the SSH User Name) to connect to the remote hosts for copying the files. ■ passphrase - Use the specified passphrase (associated with the SSH User Name) along with the SSH Key to connect to the remote hosts. ■ default - Connect to the remote hosts without a user name, password, or passphrase. This is only applicable if you are copying files to a remote host (meaning that at least one managed server is on a different physical host than the admin server host).

Table 5–5 (Cont.) Installation Properties File Parameter Reference

Parameter Name	Description
input.ssh.username	Specify the SSH user name to connect to the remote hosts. This is only applicable if you are copying files to a remote host (meaning that at least one managed server is on a different physical host than the admin server host).
input.ssh.username.alias	Specify the alias name associated with the SSH user name. This is used to store or retrieve the SSH credentials (SSH username, and either SSH password or passphrase) to and from the Oracle Wallet. This is only applicable if you are copying files to a remote host (meaning that at least one managed server is on a different physical host than the admin server host).
input.ssh.keyfile	Enter the location of the SSH key file. When left blank, the installer retrieves the file from **${user.home}/.ssh/id_dsa** directory, where **user.home** is your home directory. To use this default location, ensure that you have the private DSA key stored at this location. This is only applicable if you are copying files to a remote host (meaning that at least one managed server is on a different physical host than the admin server host).
Application Server Information	
input.target.server.name	Specify the cluster or managed application server names where you want to install the RPAS Fusion Client.
User Information	
input.security.user= [user1,user2,user3, and so on.] input.security.group= [group1,group2,group3, and so on.]	Enter enterprise users or groups for SSO, LDAP/WebLogic (supported), and WebLogic attached (for demos/development) authentication. These users and groups refer to the User IDs and user groups that are created in the ID store. The users (user or group names) you list here inform the installer which users and user groups are allowed to access the Fusion Client. The users listed in the SSO ID store must be users of the RPAS solution deployed on the RPAS Fusion Client. To support SSO, the Web deployment descriptors need to be configured to allow access pages to SSO-authenticated requests. During the application installation, the names entered here are added to the Web deployment descriptor file (`weblogic.xml`) as <principal-user> under <security-role-assignment> tag. **Note**: Using group names is preferred over user names as it allows you to add new users to the group without making changes to the deployment descriptor. The RPAS Fusion Client application roles are mapped to the enterprise roles or groups in this deployment descriptor file. For more information on setting up SSO, refer to the Appendix: Oracle Single Sign-On (SSO).
Application Deplormation	
input.app.name	Specify an application name. The RPAS Fusion Client will be deployed over the WebLogic Server with this name.
input.app.context.root	Specify the context root for the application. Once deployed, the RPAS Fusion Client will be available on the Web browser using this context path. For example, in case you set the context root to rav, you can access the application using the URL: `http://<hostname>:<port>/rav`.

Table 5–5 (Cont.) Installation Properties File Parameter Reference

Parameter Name	Description
input.app.image.repository	Specify the location or a network path where the images used in the application are located. A directory named **repository** must be created under the path specified for the input.app.image.repository. The images reside in the repository directory. Without the repository directory under the path specified in input.app.image.repository you will be unable to add/view repository images. You may create sub-directories under the repository directory, to store the images. In the Fusion Client and in the measure, the File Path starts at repository. For example, you may enter `/repository/images/my_image.jpg` in the File Path text box.
RPAS Information	
input.rpas.solution.details.known	Specify whether you know the details of the RPAS solution infrastructure and domain.
input.rpas.solution.id	The RPAS Solution ID for your RPAS application.
input.rpas.solution.desc	Specify the RPAS solution description which is displayed in the fusion client to let you select the solution.
input.rpas.server.name	Specifies the host name of the DomainDaemon from the RPAS Server installation.
input.rpas.server.port	Specifies the port number of the DomainDaemon from the RPAS Server installation.
input.rpas.domain.path	Specify the location where the RPAS domain is installed.
Automatic creation of SSL certificates	
input.useSelfSignedRootCertificate	The process of creating and configuring SSL certificates can be automated in order to cater to the common case where all the deployed RPAS components are under the customer's control. This property (when set to Yes) triggers automatic creation of SSL certificates signed by a private certificate authority. At the end of the process, the customer only has to copy the server wallet directory to its correct destinations. Use this option only if deploying to a trusted hardware and software environment that is under the control of a single business entity. Where the RPAS client or the server is not in the customer's control, externally signed SSL certificates are recommended. For additional information, refer to the section, Setup SSL.
Plug-ins (optional)	
input.homepage.module	The plug-in module shown on the Fusion Client UI home page. Module name needs to be prefixed by the name of the bundle that it belongs to: input.homepage.module=<bundlename>:<module Name> For example: input.homepage.module=aip:dashboard
input.installed.bundles	The comma-separated list of names of plug-in bundles that need to be associated with the Fusion Client (the bundles are expected to have been previously installed). This information needs to be obtained from the bundle packages: input.installed.bundles=<list> For example: input.installed.bundles=aip,poview

Table 5–5 (Cont.) Installation Properties File Parameter Reference

Parameter Name	Description
input.installed.bundles.shared.libs	List of names of the WebLogic shared libraries that the bundles will be deployed as, on the WebLogic server The order of entries of the installed bundles and installed shared libraries is important. A bundle in a particular position in the former property will receive the library name that it is in the same position in the latter property. When selecting names for installed bundles and installed shared libraries, consider naming them after the bundle name (for ease of remembering which bundle a particular shared library corresponds to). For example: input.installed.bundles.shared.libs = aipdashboard.shared.lib, poview.shared.lib
BI Reports/MDS	
input.mds.isConfigured	Whether to use an existing MDS repository or have the installer create a new one
input.mds.repository.type	The repository type. Use either File or DB. File is the default.
input.mds.repository.name	The name of the MDS repository.
input.mds.partition.name	The partition name. The default is `mdsrep` if File-based. The default is `pdborcl` if DB-based.
input.mds.repository.path	The file system path of the repository, applies to File-based type. The JNDI name of the repository if DB-based. If File-based: `<install-dir>/mdsrep.` If DB-based: `jdbc/mds/mdsrep`.
input.mds.dbvendor	The type of DB. Makes sense only for DB-based repository. Note: Only Oracle DBs are supported.
input.mds.hostname	The host name of the DB server.
input.mds.port	TNS listener port number for the host name.
input.mds.dbname	The database service name.
input.mds.user.alias	The database user alias. Enter the DB user and password in the installer's credential store wallet prior to installation.

Setting Up Environment Variables

This is the eighth pre-installation task, ensure that all previous pre-installation tasks are complete.

Before you start the installation, ensure that the following environment variables are set in the system:

- WEBLOGIC_DOMAIN_HOME – Location where the WebLogic domain is installed. For more information, refer to Setting Up the WebLogic Server.
- MIDDLEWARE_HOME – Location where the WebLogic server is installed. For more information, refer to Setting Up the WebLogic Server. This environmental variable is mandatory.
- ORAINST_HOME – An absolute path to the file containing the central inventory information. This file must define the `inventory_loc` and `inst_group` variables in it. The `inst_group` value should be the UNIX group for the OPatch utility.

  For example:

```
inventory_loc=/home/weblogic/oraInventory
inst_group=users
```

Note: You do not need to set JAVA_HOME or ORACLE_HOME. Regardless of your JAVA_HOME, JAVA_HOME is set by the installer to the one used by the WebLogic domain pointed by WEBLOGIC_DOMAIN_HOME. The same is true for ORACLE_HOME, which is set by the installer to the value needed during installation.

Although it is recommended that these variables be set up in relevant bash shell startup files (`.bash_profile`) of the system, you can also set up the variables using the EXPORT command at the UNIX prompt. For more information on setting up these variables in the startup files, refer to the operating system documentation.

To set up the environment variables for the current session, at the UNIX prompt type the following commands in sequence:

```
export WEBLOGIC_DOMAIN_HOME=<path where the WebLogic domain is installed>
```

```
For example, /u01/app/oracle/middleware/user_projects/domains/base_domain
```

Validating WebLogic and Oracle ADF Versions

This is the ninth pre-installation task, ensure that all previous pre-installation tasks are complete.

The OPatch path must be included in the PATH variable for the validation to run. This export is mandatory:

```
export PATH=<ORACLE_HOME>/OPatch:$PATH
```

When you launch the installer, it attempts to validate the version of WebLogic and the associated Oracle Application Development Framework (ADF) version for this release. You may want to run these validations without launching the installer in a standalone fashion prior to running the installer. There are two shell scripts that you can run to do this. In the **<installer>/common** directory, there are two files:

- `validateWls.sh` for validating the WebLogic version
- `validateAdf.sh` for validating the Oracle ADF version

You can run each of these scripts independently without any argument:

- `sh validateWls.sh`
- `sh validateAdf.sh`

If the validation is successful, the scripts print a success message: *Found supported version of WebLogic Server x.x.x*. Or, it exits with an error message that describes the issue. After addressing the issue, you may want to run the scripts again.

Creating User Credentials in an Oracle Wallet

This is the tenth and final pre-installation task, ensure that all previous pre-installation tasks are complete.

Note: This step is only required when Installing the RPAS Fusion Client in Silent Mode. If Installing the RPAS Fusion Client in Graphical or Text Mode, you can enter the credentials in the GUI.

Information such as user credentials for the RPAS Fusion Client is encrypted and stored in a secure location in the application installation directory. This location is called the Oracle Wallet.

When the installation starts, the administrative user credentials are retrieved from the Oracle Wallet based on the alias name specified in this window.

Note: Before installing the RPAS Fusion Client in silent mode, you must create user credentials in an Oracle Wallet. This step is optional when installing in text or graphical mode.

Storing Credentials

There are two kinds of credentials that the silent-mode installer demands to be stored in the wallet before it starts:

- **WebLogic admin userid and password**: Stored against the alias *wlAdmin* (or whatever alias is configured within the `ant.install.properties` file against the *input.admin.username.alias* property. username: *WebLogic admin user name*.
- **Client-side SSL keystore access password**: The userNameAlias must be *fckey*, however, the username can be anything as it is ignored. At runtime, this password is accessed from the WebLogic domain credential store and used to read the SSL key stores deployed on the Fusion Client.

The respective passwords are prompted for by the `save_credential.sh` script

Creating Credentials

The following steps describe how to create the user credentials:

1. Set and export the JAVA_HOME variable.
2. Cd to `<installer>/retail-public-security-api/bin`.
3. Run `sh save_credential.sh <userNameAlias> <username> <locationOfWalletDir>`.

   The following table describes the field options to run the script in Step 3.

Note: Credentials can be used in GUI mode if you select **Yes** to Retrieve Credentials from the wallet as shown in Figure 5–5, "Retrieve Credentials? Window".

Field Option	Description
<userNameAlias>	The keyname for which the credentials need to be stored.
`<username>`	The username to be stored in a secure credential wallet for the specified userNameAlias.

Field Option	Description
<locationOfWalletDir>	The directory where the wallet will be created. This is an optional parameter. If omitted, it creates the wallet under: <installer>/retail-public-security-api/secure-credential-wallet.

Installation Tasks

After you have completed the steps in the Pre-Installation Tasks section, you can start installing the RPAS Fusion Client.

> **Note:** Install any Solution Plug-ins first. For more information, refer to Appendix: Installing Solution Plug-ins or the installation instructions for the specific Oracle Retail application.

The RPAS Fusion Client Installation media includes an Oracle installer that you must run to install the RPAS Fusion Client. The installer installs the application based on the parameters specified in an installation properties file.

You can install the application in either of these modes:

- Installing the RPAS Fusion Client in Silent Mode - In silent mode, the installer processes the values set in the properties file with no manual intervention required.
- Installing the RPAS Fusion Client in Graphical or Text Mode- In graphical or text mode, the Oracle Installer will prompt you to enter or modify the value of properties specified in the installation properties file.

Installing the RPAS Fusion Client in Silent Mode

This section describes how to install RPAS Fusion Client in silent mode. Silent mode is non-interactive.

> **Note:** If you are reinstalling the Fusion Client after installing an RPAS application, you must backup the `rgbu_planning_home/Help/ohwconfig.xml` file and restore it after the installation. Otherwise, access to the application's help files is lost.

To install RPAS Fusion Client in silent mode:

1. Ensure that you have completed Setting Up Your Installation Properties File.For more information on the parameters, refer to "Installation Properties File Parameter Reference" on page E-4..

> **Note:** Ensure that the `ant.install.properties` file is available in the same directory with the `install.sh` script.

Note: This product's installer includes Ant.If Ant is already installed on your system and is version 1.6.5 or earlier, then you must run `unset ANT_HOME` on the command line to ensure that the installer uses the included version. The `unset ANT_HOME` command must be run before `./install.sh`.

2. Ensure that you have saved these user credentials in an Oracle Wallet:
   - WebLogic domain admin user credential (the entry alias is configurable)
   - Client-side SSL keystore credential (the entry alias is *fckey* and it is not configurable)

   The silent mode installer does not ask for user credentials. Instead it retrieves them from the Oracle Wallet.

   For more information, refer to Creating User Credentials in an Oracle Wallet.

   Note: The installation property `input.retrieve.credentials` must be **Yes**.

3. Ensure that the RPAS Domain and WebLogic Server are running.
4. Navigate to the **RPAS Fusion Client installation** folder, enter the following command:

   ```
   ./install.sh silent
   ```

install.sh

The `install.sh` command enables you to install RPAS Fusion Client.

Syntax

```
./install.sh <mode name>
```

Arguments

Use any arguments listed as needed:

Argument	Description
a.	Argument
b.	Description
c.	<mode name>
d.	Use this argument to specify the installation mode. You can specify the following: ■ swing – to launch a graphical installer. This is the default installation mode. In case you do not specify a mode, the installer defaults to swing mode. ■ text – to launch the installer with instructions that appear as text on the window. ■ silent – to start the installation based on the parameters set up in the `ant.install.properties` file. No manual intervention is required.

Output

The RPAS Fusion Client installation creates the application directory structure, populates it with appropriate files, and when the installation finishes, it generates a log file and two properties files.

Installing the RPAS Fusion Client in Graphical or Text Mode

If you prefer to use a guided user interface (GUI), you can use the Oracle Installer in the graphical or text mode. Although this section describes how you can install the RPAS Fusion Client in graphical mode, the same on-screen instructions appear as text instructions in the text mode.

Note: Although you do not have to set up the installation properties file when installing in graphical or text mode, it is helpful. If you set up values in the properties file, those values will be the default values in the graphical or text mode dialog.

For instructions on setting up this file, refer to Setting Up Your Installation Properties File.

If you are reinstalling the Fusion Client after installing an RPAS application, you must backup the `rgbu_planning_home/Help/ohwconfig.xml` file and restore it after the installation. Otherwise, access to the application's help files is lost.

Note: This product's installer includes Ant.If Ant is already installed on your system and is version 1.6.5 or earlier, then you must run `unset ANT_HOME` on the command line to ensure that the installer uses the included version. The `unset ANT_HOME` command must be run before `./install.sh`.

To install RPAS Fusion Client using the GUI:

1. Ensure that the RPAS Domain and WebLogic server are running.
2. If you are viewing the installer from a Windows client:
   - On the Windows client, start an Xserver program that enables you to emulate the X terminal.
   - On the application server, set the display for the Windows client where you want the Oracle Installer to display as follows:

     ```
     export DISPLAY=<IP address>:0.0
     ```
3. From your application server machine, enter the following command:

   ```
   ./install.sh
   ```

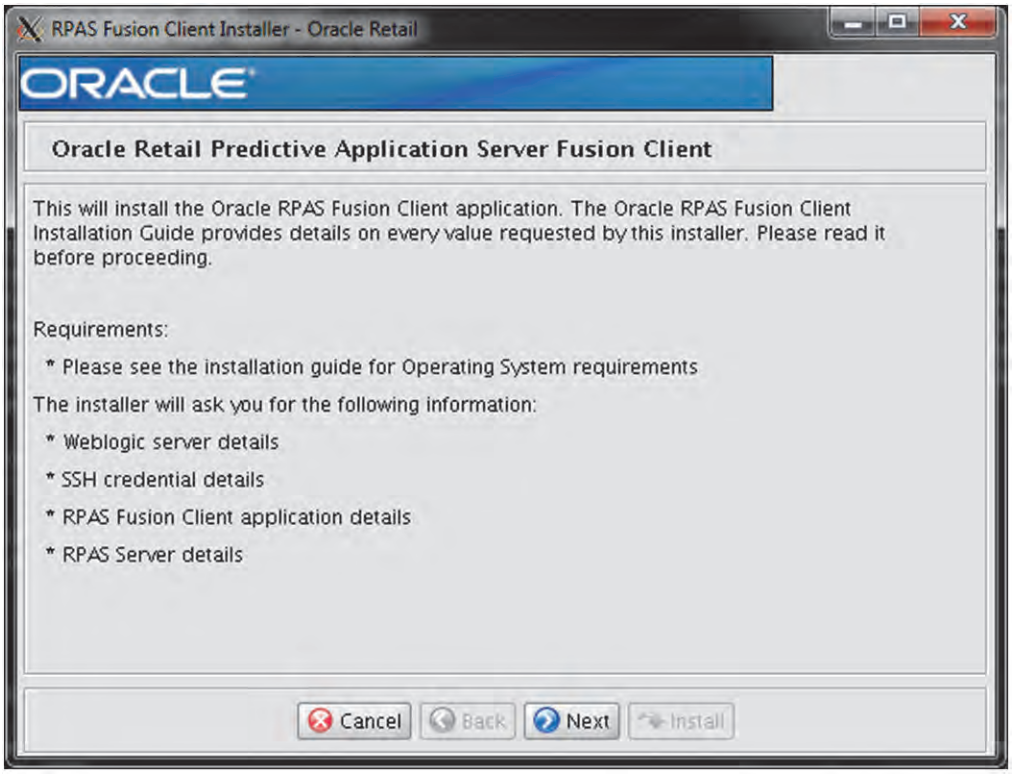
   When the installer runs, it attempts to validate the version of WebLogic and the associated Oracle ADF that are used by the WebLogic domain. If you encounter a validation error, fix the problem and try again.

   Note: If you want, you can skip these validations by setting and exporting `SKIP_FC_VALIDATION=1` before running the installer.

For more information about the `install.sh` command refer to the section, Installing the RPAS Fusion Client in Silent Mode.

4. The Oracle Retail Predictive Application Server Fusion Client Window opens. Click **Next**.

Figure 5–4 Oracle Retail Predictive Application Server Fusion Client Window



5. The Retrieve Credentials? Window opens.

Figure 5–5 Retrieve Credentials? Window

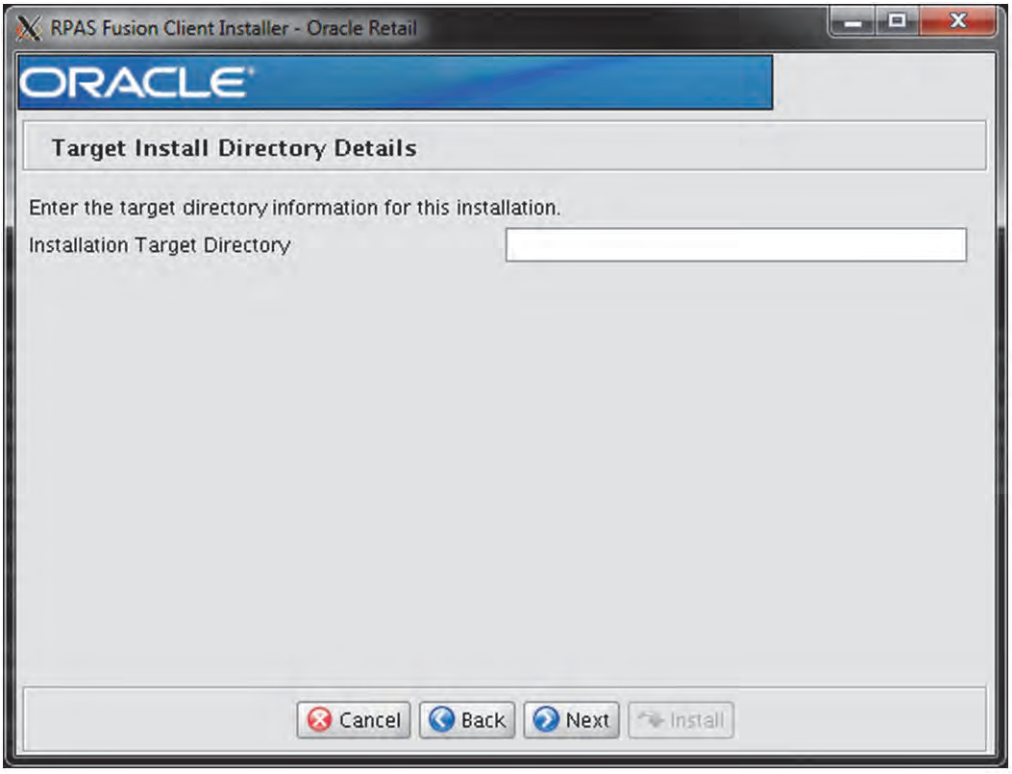
Select whether to retrieve user credentials from the Oracle Wallet:

- Select **Yes** to indicate that the installer will read the user credentials from an Oracle Wallet
- Select **No** to indicate that you will enter the user credentials in the user interface.

Click **Next**.

6. The Target Install Directory Details Window opens.

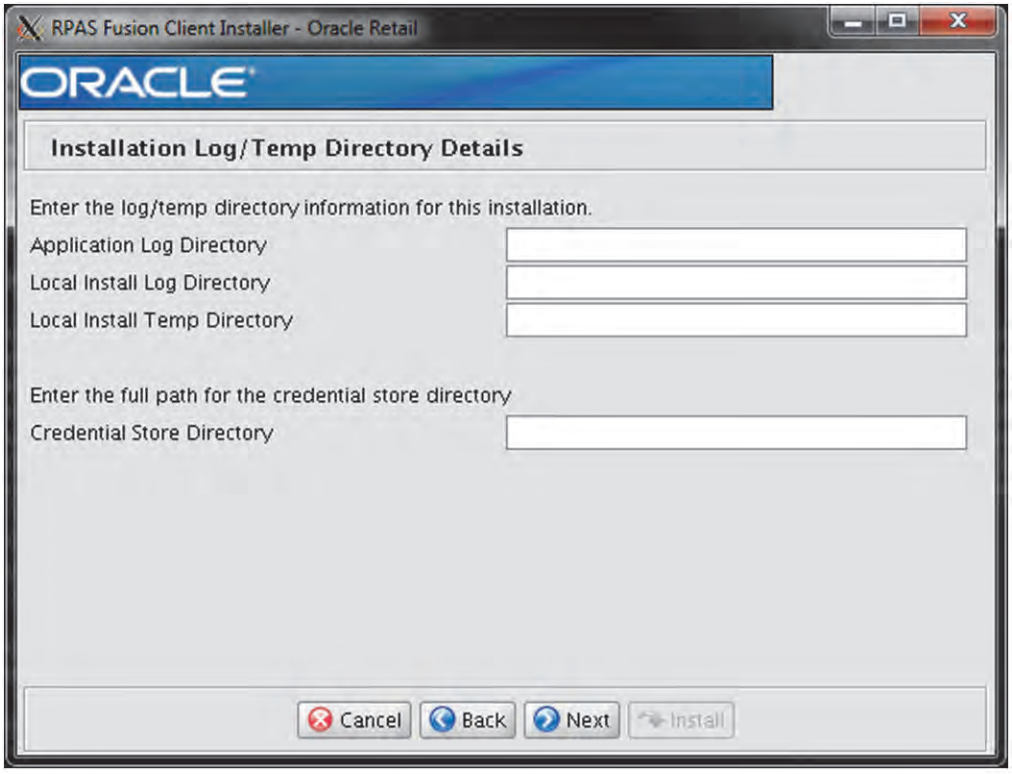
Figure 5–6 Target Install Directory Details Window



In the Installation Target Directory field, specify the location where you want to install the RPAS Fusion Client. Click **Next**.

7. The Installation Log/Temp Directory Details Window opens.

Figure 5–7 Installation Log/Temp Directory Details Window



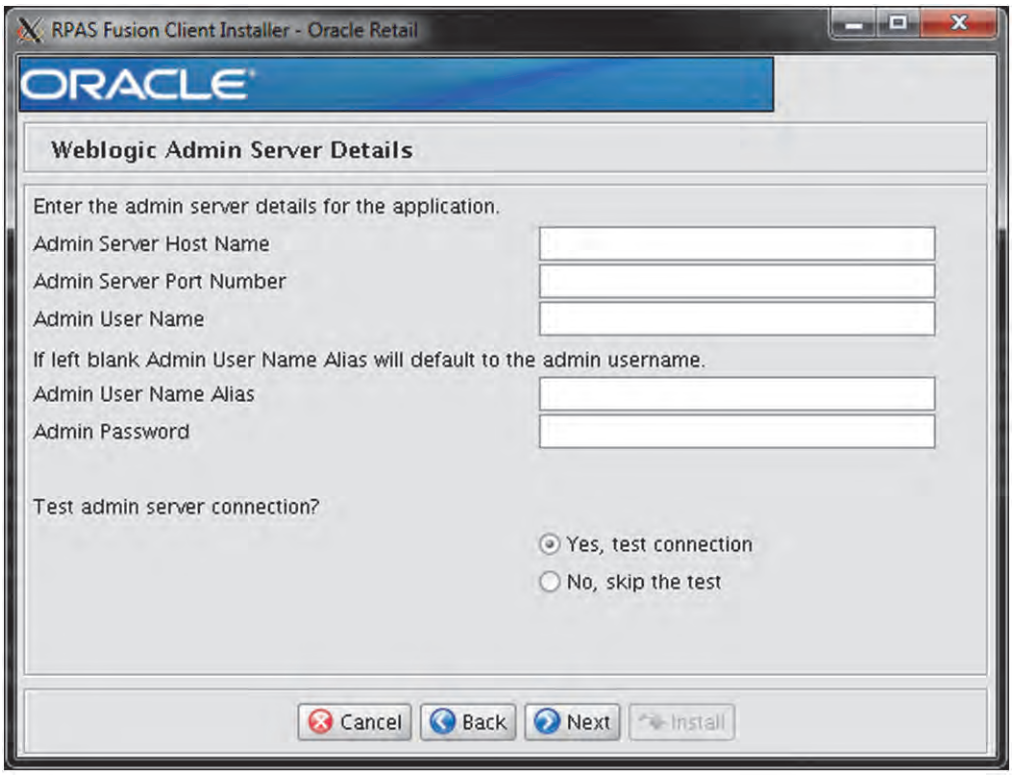
Enter the relevant information in the following fields and click **Next**:

Field	Description
Application Log Directory	Specify the location for the application log files.
Local Install Log Directory	Specify the location for the installation log files.
Local Install Temp Directory	Specify the location for the temporary file directory used during installation.
Credential Store Directory	Specify the location of the Oracle Wallet you want to use to save or retrieve user credentials. This cannot be in or under the same temporary directory named in the Local Install Temp Directory field.

Note: By default, all fields (except Credential Store Directory) are pre-populated based on the installation directory you specified in the Target Install Directory Details Window.

8. The WebLogic Admin Server Details Window opens.

Figure 5–8 WebLogic Admin Server Details Window



Enter appropriate information for the following fields and click **Next**:

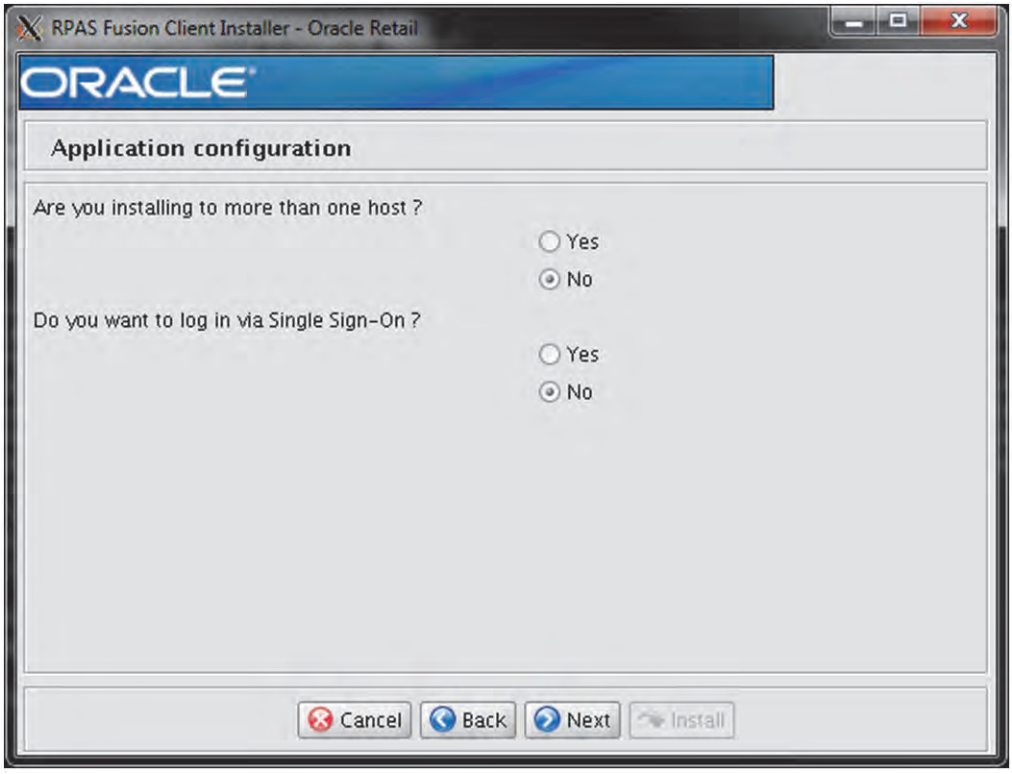
Field	Description
Admin Server Host Name	Specify the host name of the application server.
Admin Server Port Number	Specify the port number associated with the application server.
Admin User Name	Specify the administrative user name for the application server.
Admin User Name Alias	Specify an alias name for the administrative user. Specifying this name enhances the security for the application. When left blank, the alias name defaults to the administrative user name. Refer to the note following this table.
Admin Password	Specify the password associated with administrative user name.
Test admin server connection?	Select **Yes** if you want to test the connection to the admin server. Select **No** if you want to advance to the next window without testing the connection

Note: Information such as user credentials for the RPAS Fusion Client is encrypted and stored in a secure location in the application installation directory. This location is called the Oracle Wallet.

When the installation starts, the administrative user credentials are retrieved from the Oracle Wallet based on the alias name specified in this window.

9. The Application Configuration Window opens.

Figure 5–9 Application Configuration Window



a. Are you installing to more than one host?

Option	Description
Yes	This indicates an installation where at least one managed server is running on a remote machine (remote to the admin server machine). Go to Step 10.
No	This indicate an installation where all of the managed servers for deploying the Fusion Client are running on the same machine as the admin server machine, or if you are deploying to the admin server only. Go to Step 11.

b. Do you want to log in via Single Sign-On?

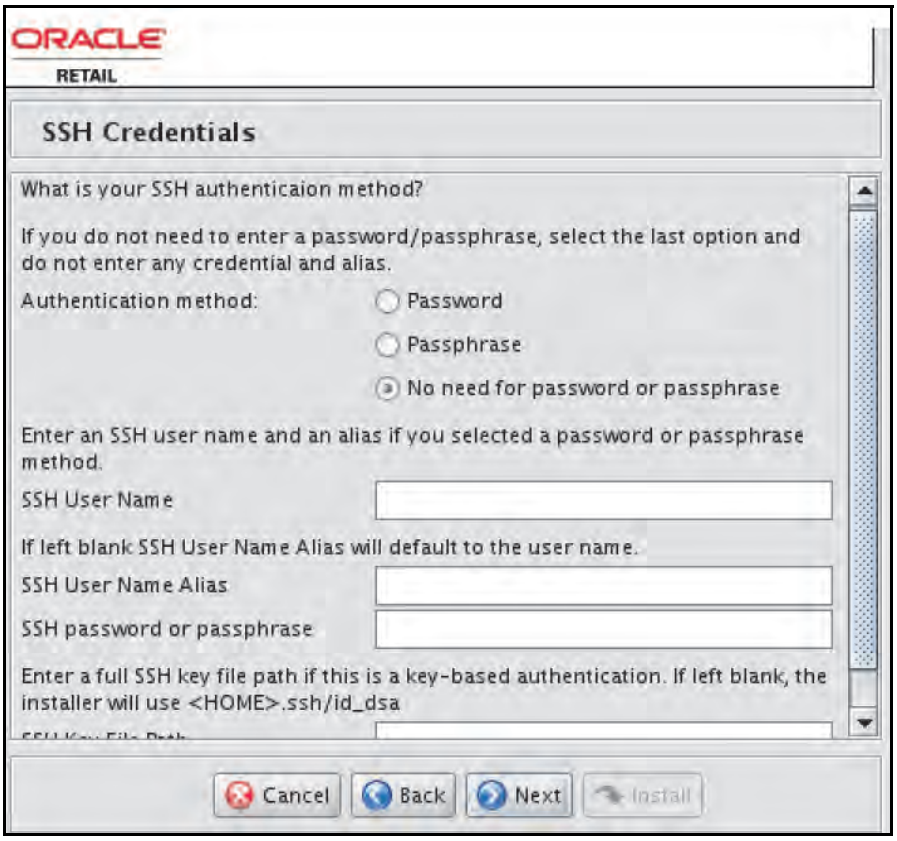
- Select **Yes** to indicate that you want use this feature. Go to Step 16.
- Click **Next**.

Note: For more information on the setting up SSO, refer to the section, Set Up Single Sign-On (SSO).

---

10. The SSH Credentials Window opens if you selected **Yes** on the Application Configuration Window, otherwise continue to Step 11

Figure 5–10 SSH Credentials Window



Enter the relevant information in the following fields and click **Next**:

Field	Description
Authentication method	Select one of the following authentication methods: ▪ **Password** - Use the specified password (associated with the SSH User Name) to connect to the remote hosts for copying the files. ▪ **Passphrase** - Use the specified passphrase (associated with the SSH User Name) along with the SSH Key to connect to the remote hosts. ▪ **No need for password or passphrase** - Default option; connect to the remote hosts without a user name, password, or passphrase.
SSH User Name	Specify the SSH user name to connect to the remote hosts.
SSH User Name Alias	Specify the alias name associated with the SSH user name. Specifying an alias name enhances the security for the application. When left blank, the alias name will default to the administrative user name. Refer to the first note following this table.
SSH password or passphrase	Based on the authentication method you selected, enter the relevant SSH password or passphrase.
SSH Key File Path	In case you selected the Passphrase option in the Authentication Method field, enter the location of the SSH key file. When left blank, the installer will retrieve the file from **${user.home}/.ssh/id_dsa** directory, where **user.home** is your home directory. To use this default location, ensure that you have the private DSA key stored at this location. Refer to the second note following this table.

Note: Information such as user credentials for the RPAS Fusion Client is encrypted and stored in a secure location in the application installation directory. This location is called the Oracle Wallet.

When the installation starts, the administrative user credentials are retrieved from the Oracle Wallet based on the alias name specified in this window.

Note: The **SSH User Name** and **SSH password or passphrase** fields do not appear when you choose to retrieve the existing SSH credentials (the **Yes, retrieve saved credentials** option in the Retrieve Credentials? Window).

The existing SSH user credentials will be retrieved based on the alias name for the SSH user.

11. The Application Server Details Window opens.

Figure 5–11 Application Server Details Window

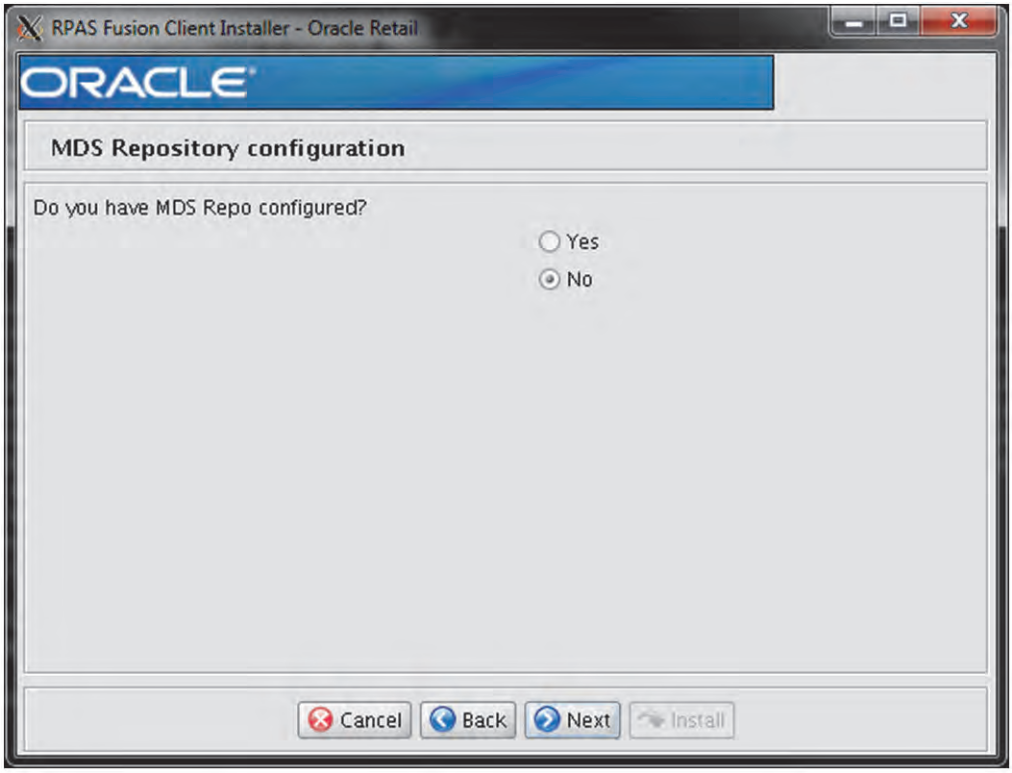
Enter relevant information for the following fields and click **Next**:.

Field	Description
Cluster or Managed Server Names, comma-separated	Enter the cluster or managed server names where you want to deploy the fusion client.

Field	Description
Client keystore password	Enter the client keystore password which will be used for client authentication. For additional information about the client keystore password, refer to Setting Up a Trust Store in the JKS Format.

12. The MDS Repository Configuration Window opens.

Figure 5–12 MDS Repository Configuration Window



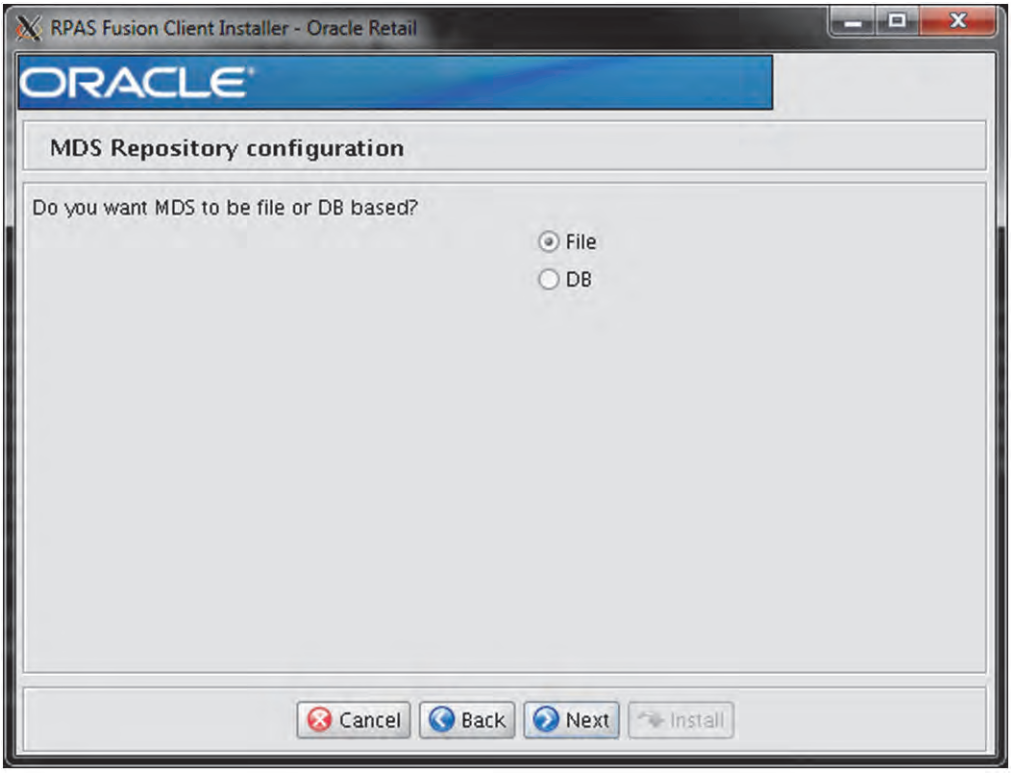
Select one of the following options and click **Next**:

Note: It is strongly recommended to let the RPAS installer create an MDS repository using the default values as it greatly simplifies the process of installation.

Option	Description
Yes	This indicates you have already set up the MDS repository. Go to Step 16.
No	This indicates you wish the RPAS installer to create an MDS repository. Go to Step 13.

13. The 2nd MDS Repository Configuration Window opens.

Figure 5–13 2nd MDS Repository Configuration Window

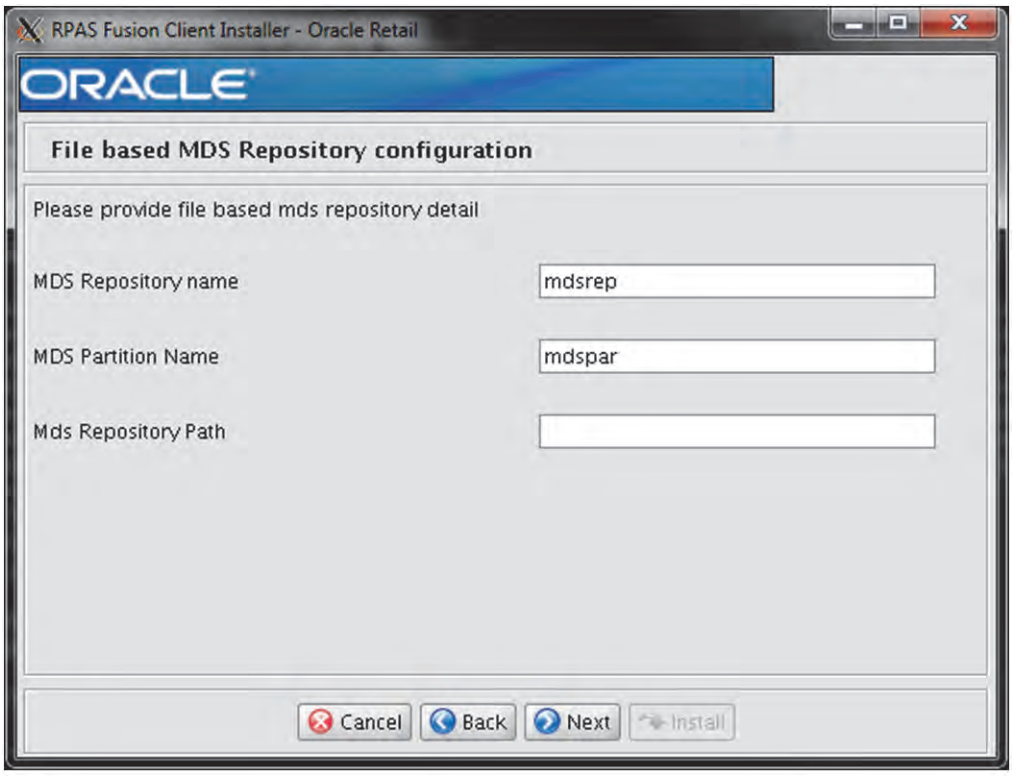


Select one of the following options and click **Next**:

Option	Description
File	This indicates you wish to set up the MDS repository as a file based repository. Go to Step 14.
DB	This indicates you wish to set up the MDS repository as a data base based repository. Go to Step b.

14. Either an existing file based repository is used, or a new one has to be created.The File Based MDS Repository Configuration Window opens.

Figure 5–14 File Based MDS Repository Configuration Window



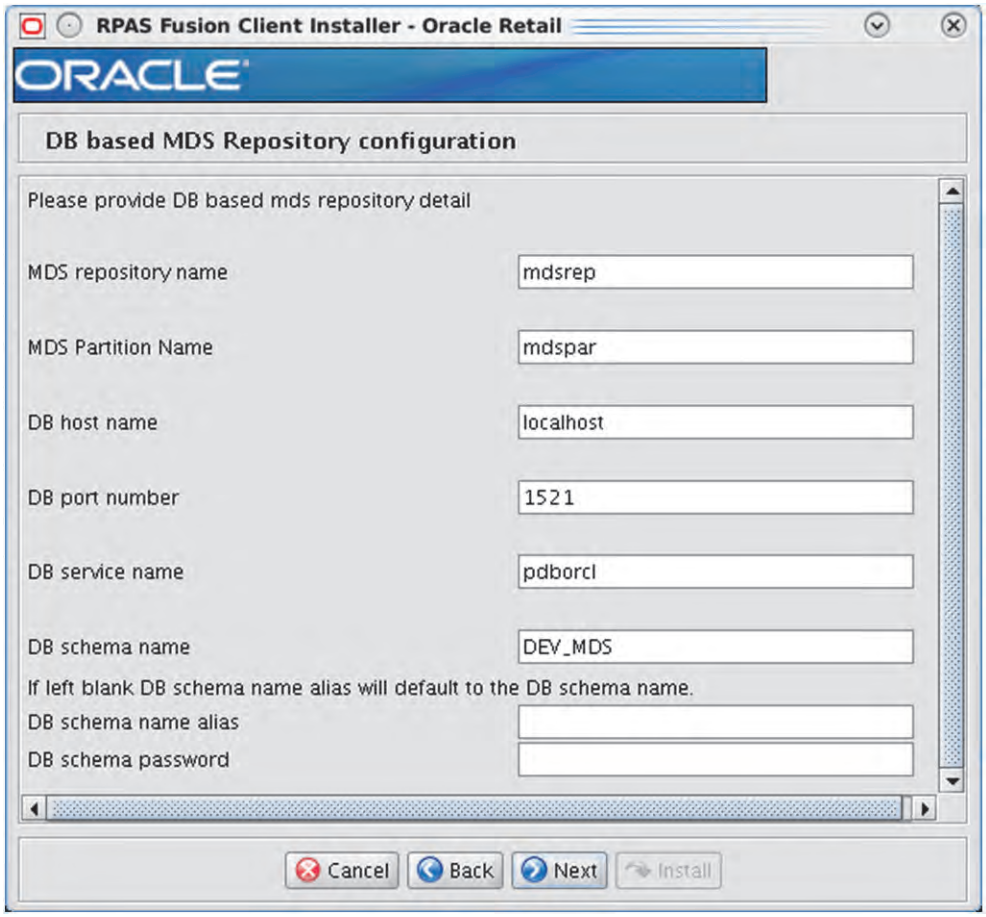
Enter relevant information for the following fields and click **Next**:

Field	Default Value
MDS Repository Name	mdsrep
MDS Partition Name	mdspar
MDS Repository Path	<install-dir>/mdsrep

15. Either an existing DB based repository is used, or a new one has to be created.The DB Based MDS Repository Configuration Window opens.

Note: A DB schema must be created beforehand using the RCU as described in the sections "Running the RCU" and "Creating Schemas." Ensure that the option for **Metadata Services** is selected in the RCU.

Figure 5–15 DB Based MDS Repository Configuration Window



a. To create a new DB based MDS Repository, enter relevant information for the following fields and click **Next**:

Field	Default Value
MDS Repository Name	mdsrep
MDS Partition Name	mdspar
DB Host name	localhost
DB Port Number	
DB Service Name	
DB Schema Name	DEV_MDS
DB Schema Name Alias	If left blank, it defaults to the DB schema name
DB Schema Password	

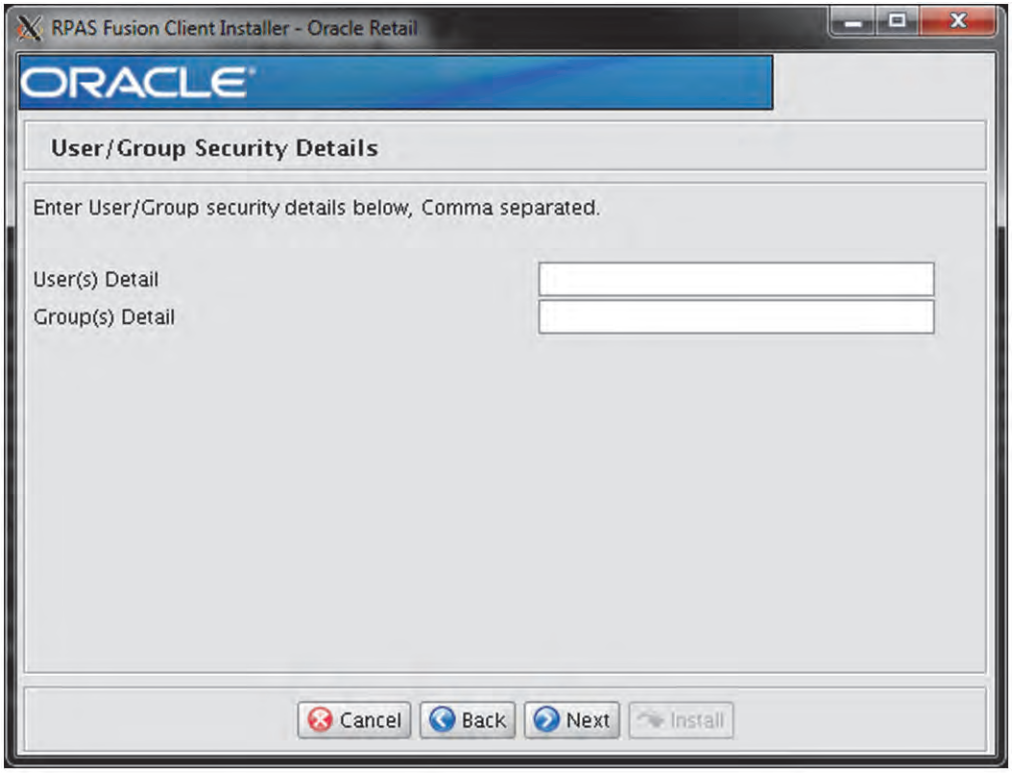
b. To update an existing DB based MDS Repository, only enter information for the following fields and click **Next**.

Note: The MDS Repository Name is as specified while creating the repository, but without the "mds-" prefix that gets automatically added while creating it. For example, suppose you created an MDS repository called "mdsrep2" using the Enterprise Manager tool. Enterprise Manager automatically generates the repository name as "mds-mdsrep2". Here you enter the value as "mdsrep2", that is to say you skip the prefix.

Field	Default Value
MDS Repository Name	mdsrep
MDS Partition Name	mdspar
MDS repository JNDI name	jdbc/mds/mdsrep **Note:** This has to match the JNDI name that was assigned to the MDS repository.

16. The User/Group Security Details Window opens.

Figure 5–16 User/Group Security Details Window



Enter relevant information for the following fields and click **Next**:

Note: Users and Groups are processed by the installer only if using a file-based policy store.

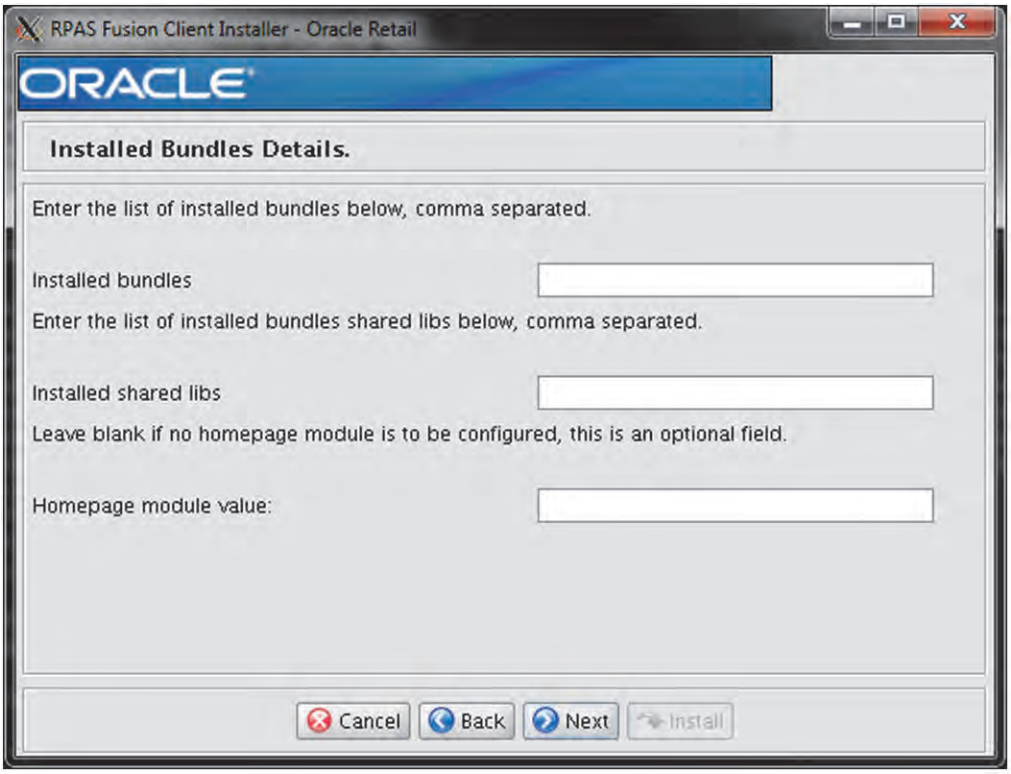
For guidance on setting up an OID-based policy store, refer to the chapter "Configuring the OPSS Security Store" in the Oracle® Fusion Middleware Application Security Guide 11g Release 1 (11.1.1) Configuring the OPSS Security Store. This document is available through My Oracle Support.

If using an OID-based policy store, then user/group information entered here is not used. See Upload Application Security Policies to OID-based Domain Policy Store for instructions.

Field	Description
User(s) Detail	Enter all user names; each separated by a comma. input.security.user= [user1,user2,user3, and so on.]
Group(s) Detail	Enter all user group names; each separated by a comma. input.security.group= [group1,group2,group3, and so on.]

17. The Installed Bundles Details Window opens.

Figure 5–17 Installed Bundles Details Window



Note: The order of entries of the installed bundles (also known as solution plug-ins or functional modules) and installed shared libraries is important. A bundle in a particular position in the former property will receive the library name that it is in the same position in the latter property.

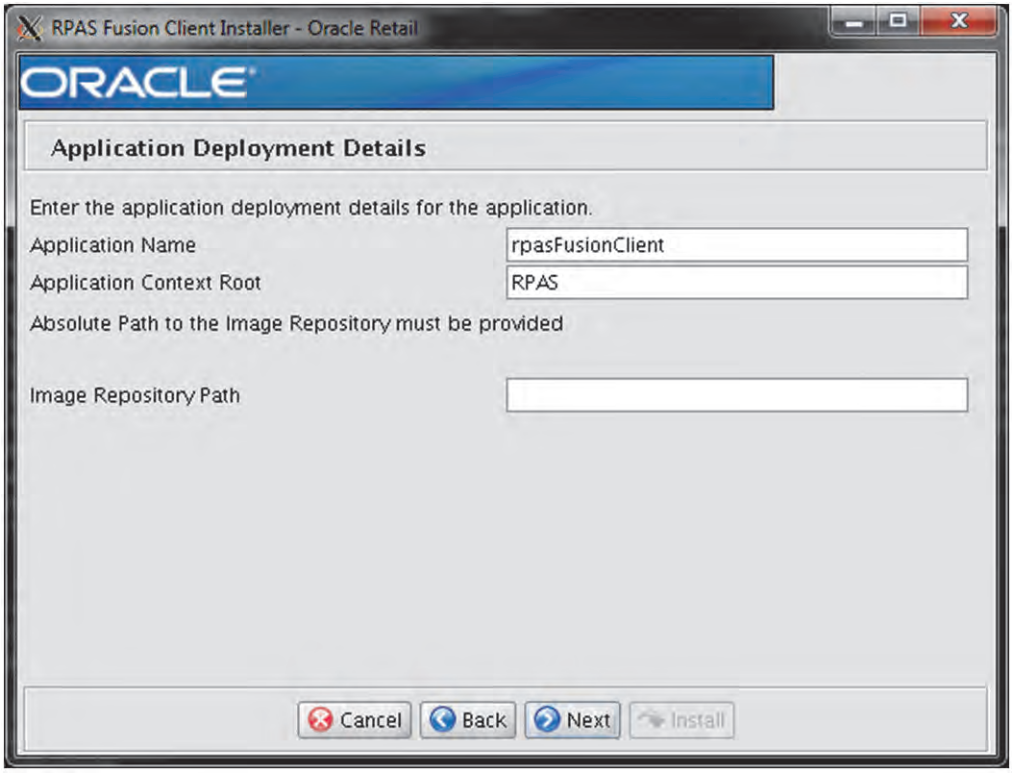
When selecting names for installed bundles and installed shared libraries, consider naming them after the bundle name (for ease of remembering which bundle a particular shared library corresponds to). For example: input.installed.bundles.shared.libs = aipdashboard.shared.lib, poview.shared.lib

Enter relevant information for the following fields and click **Next**:

Field	Description
Installed bundles	Enter the list of installed bundles (also known as solution plug-ins or functional modules); each separated by a comma. For example:aip,poview
Installed shared libs	Enter the list of installed shared libraries each separated by a comma. For example: aipdashboard.shared.lib, poview.shared.lib
Homepage Module value	Enter any homepage modules For example: aip:dashboard

18. The Application Deployment Details Window opens.

Figure 5–18 Application Deployment Details Window

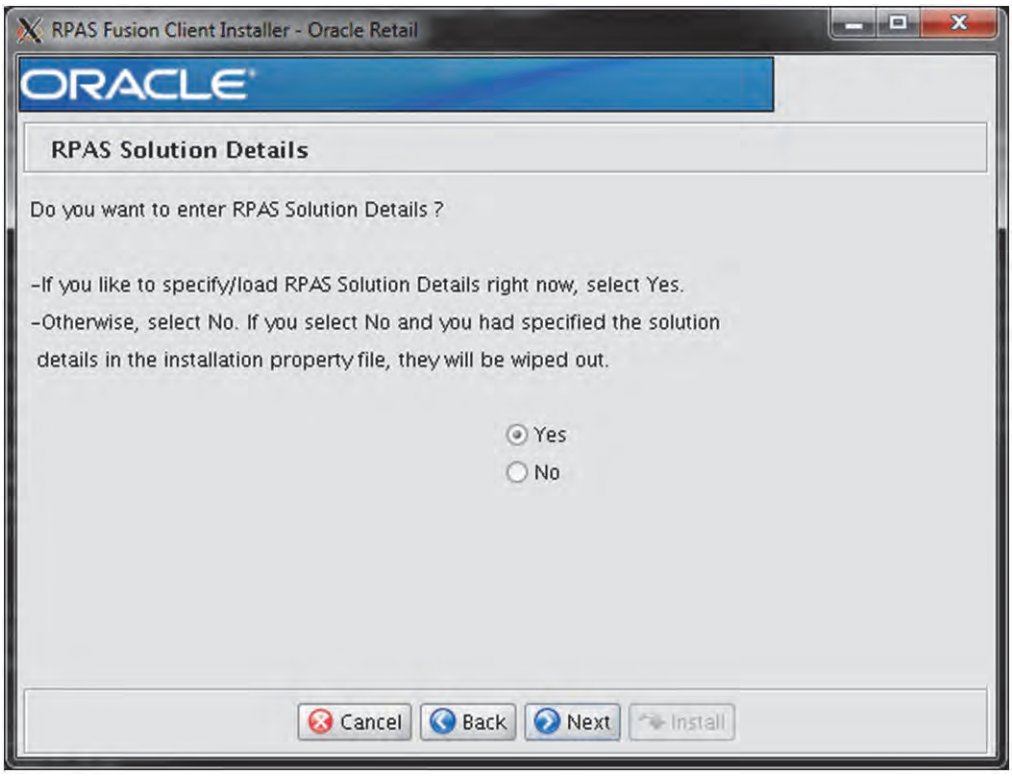


Enter relevant information for the following fields and click **Next**:

Field	Description
Application Name	Specify an application name. The RPAS Fusion Client will be deployed over the WebLogic Server with this name.
Application Context Root	Specify the context root for the application. Once deployed, the RPAS Fusion Client will be available on the Web browser using this context path. For example, if you set the context root to rav, you can access the application using the URL: `http://<hostname>:<port>/rav`.
Absolute Path to the Image Repository	Specify the location or a network path where the images used in the application are located. Images should be located in the repository folder under the specified path.

19. The RPAS Solution Details Window opens.

Figure 5–19 RPAS Solution Details Window

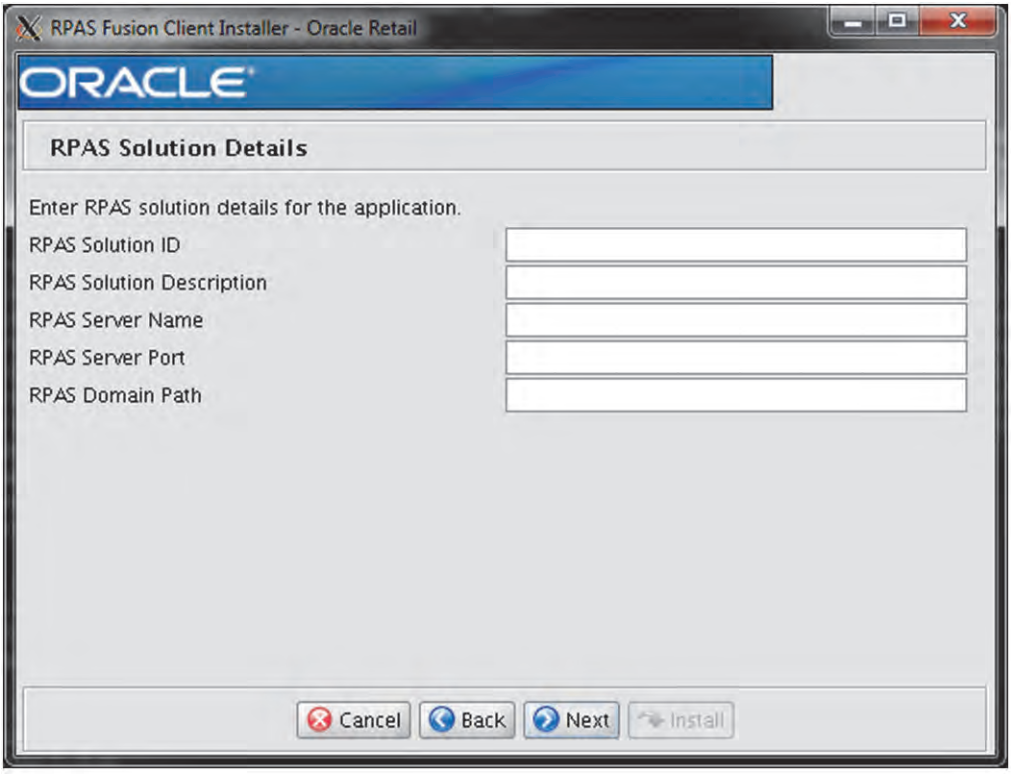


Select one of the following options and click **Next**:

- Select **Yes** to enter RPAS Server and Domain details in the Installer. During the installation, the relevant RPAS Fusion Client configuration files will be updated based on the information you enter here. Go to Step 20.
- Select **No** to skip adding the RPAS Server and Domain details and add them later manually. For more information on adding this configuration manually, refer to the *Oracle Retail Predictive Application Server Administration Guide for the Fusion Client*. Go to Step 22.

20. The RPAS Solution Details (2) Window opens with fields to collect RPAS Solution information.

Figure 5–20 RPAS Solution Details (2) Window



Enter relevant information for the following fields and click **Next**.

Field	Description
RPAS Solution ID	Specify the RPAS Solution ID for your RPAS application.
RPAS Solution Description	Specify the RPAS Solution Description which is displayed within the RPAS Fusion Client.
RPAS Server Name	Specify the host name where the RPAS infrastructure is installed.
RPAS Server Port	Specify the post associated with the RPAS installation.
RPAS Domain Path	Specify the location where the RPAS domain is installed. **Note:** This must be a simple domain or a global domain. You should not specify a local domain within a global domain environment.

21. The Self-sign Root Certificate? Window opens.

Figure 5–21 Self-sign Root Certificate? Window

Select one of the following options and click **Next**:

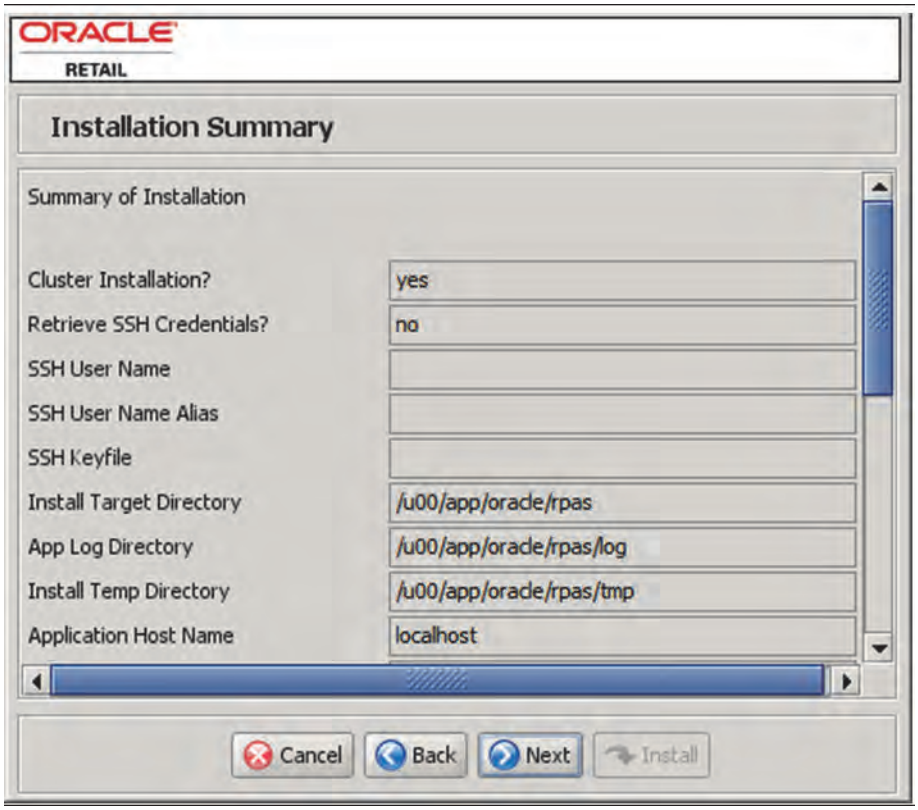
- If set to **Yes**, this property triggers automatic creation of SSL certificates signed by a private certificate authority and stores them in wallets. At the end of the process, copy the server wallet directory to its correct destinations.
- If set to **No**, you will need to manually create and configure SSL certificates and store them in wallets before or after the installation.

Note: Refer to "Setup SSL" for information about manual deployment of the SSL certificates setup.

---

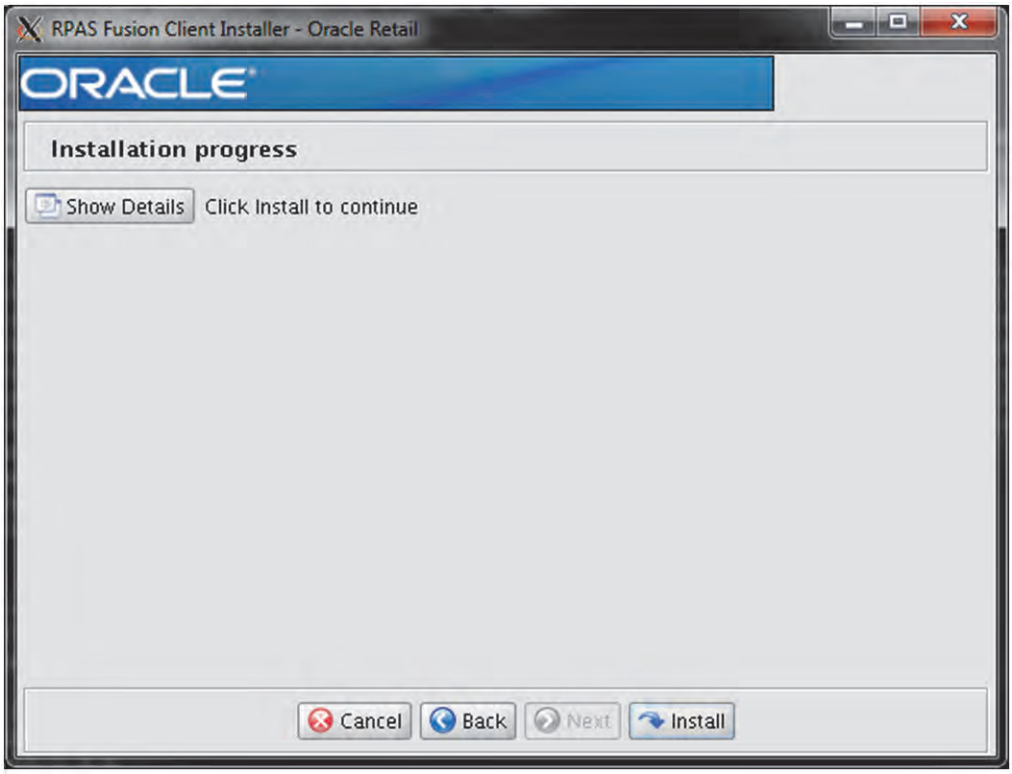
22. The Installation Summary Window opens. Review the installation summary and click **Next**.

Figure 5–22 Installation Summary Window



23. The Installation Progress Window opens. Click **Install** to start the installation.

Figure 5–23 Installation Progress Window



24. After the installation is complete, click **Exit** to close the Installer.
25. Restart the WebLogic server, and then verify that the application is accessible over the network. In a Web browser, enter the following URL in the Address bar, and press **Enter**:

    `http://<hostname>:<portnumber>/<contextroot>`

    Note: In the previously listed URL, `<hostname>`, `<portnumber>`, and `<contextroot>` represent the host name, port, and context root you set up for the application during the installation. You must specify the relevant values in the Address bar.

Postinstallation Tasks

Before you log on to the application, you must set up the Fusion Client based on your business need. This includes the following tasks:

- Multi-Solution Taskflow and Resource Properties Setup
- Configuring External Authentication
- Setup SSL
- OBIEE Report Configuration Tasks
- Clear the Browser Cache
- Review the RPAS Configuration Property Files (optional)
- Set Up Single Sign-On (SSO) (optional)

- Oracle Linux on x86 Architecture
- Upload Application Security Policies to OID-based Domain Policy Store
- Enable Gzip Compression
- Review any Troubleshooting issues
- Set up the configuration properties file, pivot table styles, and layout and formatting. For more information, refer to the *Oracle Retail Predictive Application Server Administration Guide for the Fusion Client*.
- Set up the RPAS solution to work with the RPAS Fusion Client. For more information, refer to the installation guide of the relevant RPAS solution.

Multi-Solution Taskflow and Resource Properties Setup

The RPAS Fusion Client provides a flexible approach for you to access RPAS workbooks using a Multi-solution taskflow. The taskflow allows administrators to better describe and model their business practices within the client. Administrators can create custom taskflow elements (activity groups, activities, tasks, and steps) for workbook configuration. These taskflow elements then provide a more intuitive and business practice-oriented view of the structural elements of the RPAS domain. The taskflow is created as part of the RPAS Configuration, using the RPAS Configuration Tool and can be modified as needed. The files,`taskflow.xml` and `taskflowBundle.properties`, then get generated as part of the RPAS domain creation.

One of the key configuration and setup steps postinstallation is to create the `Taskflow_MultiSolution.xml` file and its resource properties using the `taskflow.xml` and `taskflowBundle.properties` files.

Note: For information about how to setup and configure the Multi-solution Taskflow, refer to the section, "Postinstallation Configuration" in the *Oracle Retail Predictive Application Server Administration Guide for the Fusion Client*.

For information about the Multi-solution Taskflow elements, refer to the chapter, "Creating a Multi-solution Taskflow" in the *Oracle Retail Predictive Application Server Configuration Tools User Guide*.

This is one of the key configuration steps to use the RPAS Fusion Client.

Configuring External Authentication

There are three types of external authentication, one that supports SSO and the other is standalone authentication (non-SSO). The WebLogic Server can be attached to a standalone user management system such as Oracle Internet Directory (OID), or it can be configured to use an SSO system.

The third type of external authentication also has WebLogic attached and is not used for production, but is useful for development and demonstration purposes.

Authentication (SSO)

The RPAS Fusion Client is certified for perimeter authentication using Oracle SSO 11g

Standalone Authentication (Non-SSO)

Users and groups can be set up in a user management system such as OID or even WebLogic's embedded LDAP server. The corresponding authentication provider needs to be configured within the WebLogic Server default security realm called *myrealm*.

WebLogic has out of the box support for several LDAP-based authentication systems including OID. The configuration of the authentication provider can be conveniently done within the WebLogic Admin Console. This configuration provides WebLogic with information about the user management system. For example, in the case of OID one must specify its host name and port, userid, and password needed to access it, and the root location in the LDAP directory tree where users and groups can be found. When users access the application for the first time, WebLogic redirects the user to a login page. The entered User ID and Password are validated by the authentication provider within WebLogic against the user management system.

The enterprise users and groups specified during the Fusion Client installation need to be created within the user management system.

> **Note:** For more information, refer to the WebLogic documentation for details on configuring authentication providers.

External Authentication Process

The following table provides high level process steps for configuring external authentication and whether or not it applies to SSO, non-SSO, or both.

Process Steps	Authentication Type
Create an 11g WebGate definition in Oracle Access Manager. Specify the base URL that points to the Oracle HTTP Server (OHS) instance that has the WebGate module plugged in. Ensure that certain application URLs are excluded from authorization as listed in "URL Protection Rules". **Note:** For performance and high availability reasons it may be necessary to deploy multiple OHS instances with a load balancer in front. In this case specify the base URLs for all the OHS instances in the WebGate definition.	SSO
Configure the `mod_wl_ohs` module on the OHS instance to tunnel requests to the application servers on which the Fusion Client is deployed. (Do it on each of the OHS instances, if using multiples.)	SSO
Configure the OAM identity asserter and the OID authentication provider on the default security realm on the application servers to communicate with the OID instance.	SSO
Create the same users in the Oracle Identity Management (IdM) as in RPAS domain. These are referred to as enterprise users.	SSO and non-SSO
Define an RPAS specific enterprise role or group. This assists in providing all your enterprise users with RPAS access at one go (since the user principal mapping done during the install becomes much simpler). Make all the users members of this role or group.	SSO and non-SSO
Configure an authentication provider in WebLogic. Specify sufficient as the control flag value. This provider should be of a type that is capable of interacting with the IdM. WebLogic comes with out of the box support for several IdM types, for example OID.	Non-SSO

URL Protection Rules

By default the following URLs are protected:

Protected URLs
/
/.../*

Exclude the following URLs:

URLs to Excude
/.../*.gif
/.../*.png
/.../images/*
/.../rpas_exit.jsp
/.../*.js
/.../*.css
/.../*.html

Setup SSL

This section describes two-way SSL configuration on the RPAS server.

About SSL and RPAS

Secure Sockets Layer (SSL) in a widely used cryptographic protocol designed to provide communications security over a computer network. RPAS uses SSL to protect communications between the RPAS client and server. In 14.1RPAS, SSL protocol version SSLv3 is used prior to 14.1.0.5. From 14.1.0.5, SSL protocol version TLSv1.2 is used.

RPAS supports all SSL configurations which include:

- One-way SSL
- Two-way SSL
- SSL without authentication

> **Note:** The Fusion Client only supports two-way SSL.

One-way and two-way SSL require key store, trust store, or both to manage certificates. A key store contains a private key and its corresponding public certificate chain. A trust store contains trusted public certificates and certificate chains.

The RPAS server uses Oracle Wallet for its key store and trust store. The same wallet is used for both the Key Store and Trust Store. Oracle Wallet is managed by a command line utility called `orapki` which is included in the RPAS installation.

The Fusion Client uses a Java Key Store (JKS) to act as its Key Store and Trust Store. The Certificate Store and Trust Store may be stored in separate JKS files. JKSs are managed by the keytool utility which is included in the standard JDK.

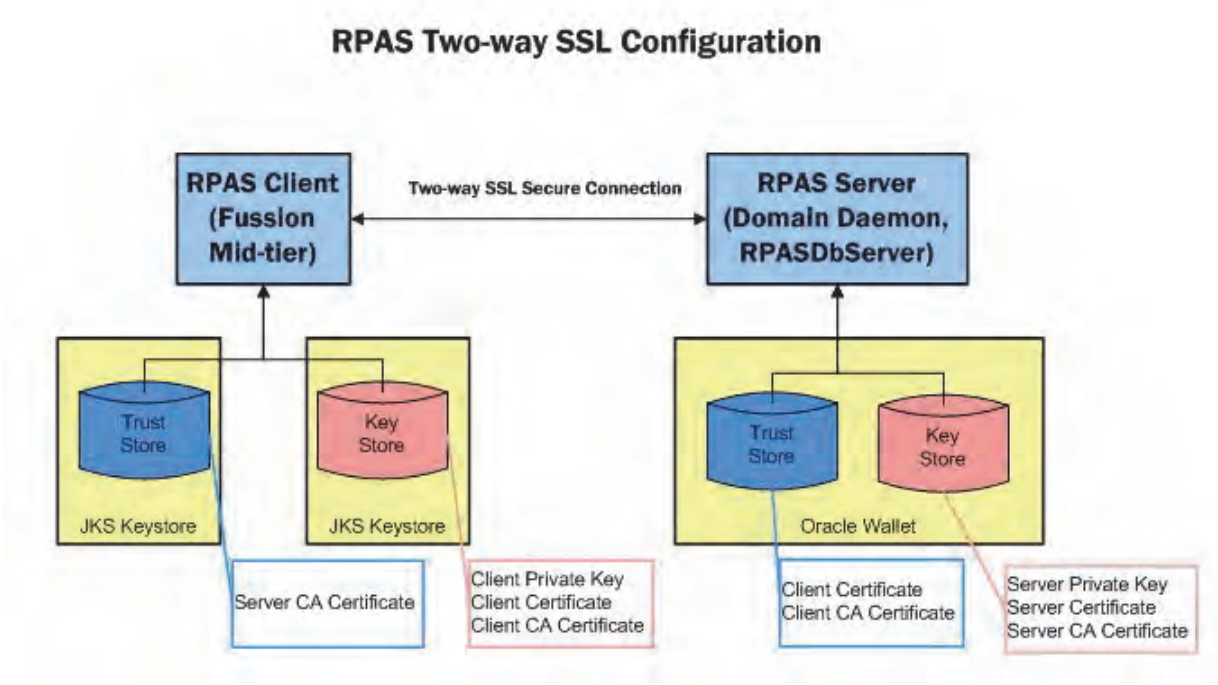
Oracle Wallet and JKS stores cannot be used interchangeably, though the `orapki` tool can be used to convert between them.

To enable SSL in RPAS, two command line options are provided for the Domain Daemon. One is for the SSL type and the other is for the Oracle Wallet location which is optional for some SSL types. When the RPAS Classic Client or the mid-tier process of the Fusion Client connects to the Domain Daemon, it detects the type of SSL for the connection and automatically chooses the appropriate handshake protocol. The client may need to configure its own certificate store in advance. RPAS DB Server inherits the SSL setup from the Domain Daemon and no additional configuration is needed.

Two-way SSL

This section describes two-way SSL as illustrated in Figure 5–24.

Figure 5–24 Configuration of Two-way SSL for RPAS



Two-way SSL authenticates both the server and the client. Both sides must have a key store and a trust store. RPAS server uses one Oracle Wallet for both stores. Two-way SSL is not supported in the Classic Client. For Fusion client mid-tier, both the trust store and the key store are in the JKS format.

The trust store must contain the CA certificate or self-signed root certificate for the other side so as to enable mutual authentication. In addition, the trust store of the server must contain the client certificate so that it can recognize and authorize that particular client.

> **Note:** For additional information on SSL, refer to the *Oracle Retail Predictive Application Server Administration Guide for the Fusion Client*.

Creating a Self-signed Root Certificate

Perform the following steps to create a self-sign root certificate. The same root certificate can be used to generate both server and client certificates.

Note: Under Cygwin environments on Windows, the path of the wallet must be in Windows format and not Cygwin format. For example, it must be:

`C:/wallets/root, not /cygdrive/c/wallets/root.`

1. Create an Oracle wallet using this format: `orapki wallet create -wallet {root_wallet_directory} -pwd {root_wallet_password}`

Example 5–2 Oracle Wallet

```
orapki wallet create -wallet C:/wallets/root -pwd rootpass1
```

2. Generate a private key and a self-signed root certificate using this format: `orapki wallet add -wallet {root_wallet_directory} -keysize {key_len} -dn {root_dn} -self_signed -validity {validity_days} -pwd {root_wallet_password}-addext_ski`

Note: Oracle OSSA requires that the minimum key size is 2048.

Example 5–3 Private Key and a Self-signed Root Certificate

```
orapki wallet add -wallet C:/wallets/root -keysize 2048 -dn "cn=rpas_qa_
ca,dc=us,dc=oracle,dc=com" -self_signed -validity 3650 -pwd rootpass1-addext_ski
```

3. Export the root certificate chain to a file for later use using this format: `orapki wallet export_trust_chain -wallet {root_wallet_directory} -certchain {root_cert_chain_file} -dn {root_dn} -pwd {root_wallet_password}`

Example 5–4 Root Certificate Chain

```
orapki wallet export_trust_chain -wallet C:/wallets/root -certchain
C:/wallets/root_chain.txt -dn "cn=rpas_qa_ca,dc=us,dc=oracle,dc=com" -pwd
rootpass1
```

Two-way SSL Setup

The process of two-way SSL configuration on the RPAS server is described in the following table.

Step	Process	Reference
1.	Set up a server key store. One Oracle wallet can act as both key and trust stores.	Setting Up a Server Key Store
2.	Import certificates.	Import Certificates
3.	Start the Domain Daemon with the appropriate options.	Start Domain Daemon with Two-way SSL Options
4.	Set up a client key store. One Oracle wallet can act as both key and trust stores.	Setting Up a Client Key Store
5.	Set up a trust store in the JKS format.	Setting Up a Trust Store in the JKS Format

Setting Up a Server Key Store

Use the following instructions to set up a server key store.

1. Create an Oracle Wallet with Auto Login using this format: `orapki wallet create -wallet {server_wallet_directory} –auto_login -pwd {server_wallet_password}`

   Auto-login means that the wallet can be read without a password. It is generally protected by file system permission. The default file permission of the wallet is 0300 which means that it is only accessible by the owner.

   Note: Oracle OSSA requires that the minimum key size is 2048.

Example 5–5 Oracle Wallet with Auto Login

```
orapki wallet create -wallet C:/wallets/server –auto_login -pwd serverpass1
```

2. Generate a private key using this format: `orapki wallet add -wallet {server_wallet_directory} -keysize {key_len} -dn {server_dn} -pwd {server_wallet_password}`

   Note: Oracle OSSA requires that the minimum key size is 2048.

Example 5–6 Private Key

```
orapki wallet add -wallet C:/wallets/server -keysize 2048 -dn "cn=rpas_qa_
server,dc=us,dc=oracle,dc=com" -pwd serverpass1
```

3. Export a certificate request using this format: `orapki wallet export -wallet {server_wallet_directory} -dn {server_dn} -request {server_cert_req_file} -pwd {server_wallet_password}`

Example 5–7 Certificate Request

```
orapki wallet export -wallet C:/wallets/server -dn "cn=rpas_qa_
server,dc=us,dc=oracle,dc=com" -request C:/wallets/server_cert_req.txt -pwd
serverpass1
```

4. Create the server certificate.

   Note: For additional information, refer to the section, "Creating a Self-signed Root Certificate".

   a. If a third-party Certificate Authority (CA) is used, send the server certificate request file ({`server_cert_req_file`}) to the CA. The CA sends back a certificate for the server (referred to as {`server_certificate_file`}) along with the public certificate of the CA (referred to as {`root_cert_chain_file`}).

   b. If the self-signed root certificate is used, run the following command:

      `orapki cert create -wallet {root_wallet_directory} -request {server_cert_req_file} -cert {server_certificate_file} -validity {validity_days} -pwd {root_wallet_password}`

Example 5–8 Self-signed Root Certificate

```
orapki cert create -wallet C:/wallets/root -request C:/wallets/server_cert_req.txt
-cert C:/wallets/server_cert.txt -validity 3650 -pwd rootpass1
```

5. Import the CA or self-signed certificate into the wallet using this format: `orapki wallet add -wallet {server_wallet_directory} -trusted_cert -cert {root_cert_chain_file} -pwd {server_wallet_password}`

Example 5–9 Import Certificate

```
orapki wallet add -wallet C:/wallets/server -trusted_cert -cert C:/wallets/root_
chain.txt -pwd serverpass1
```

> **Note:** For third-party CA, the public certificate chain may contain more than one certificate. These certificates need to be imported one by one starting from the top of the chain.

6. Import the server certificate into the wallet using this format: `orapki wallet add -wallet {server_wallet_directory} -user_cert -cert {server_certificate_file} -pwd {server_wallet_password}`

Example 5–10 Import the Server Certificate

```
orapki wallet add -wallet C:/wallets/server -user_cert -cert C:/wallets/server_
cert.txt -pwd serverpass1
```

Now the key store is ready for the server.

Import Certificates

For every client connecting to the RPAS server using two-way SSL, their certificates must be imported into the server wallet as trusted certificates. Use the following command:

```
orapki wallet add -wallet {server_wallet_directory} -trusted_cert -cert
{client_cert_file} -pwd {server_wallet_password}
```

> **Note:** If the client certificates are from different CA other than the ones already in the server wallet, the CA certificates must be imported first.

Start Domain Daemon with Two-way SSL Options

To start the Domain Daemon with two-way SSL support, use the following command line options:

```
DomainDaemon -port portNum -ssl 2 -wallet file:<walletLocation> -start
```

Where the `<walletLocation>` is the absolute path to the directory of the Oracle Wallet. It cannot be a relative path.

Example 5–11 Domain Daemon with Two-way SSL Support

```
DomainDaemon -port 12348 -ssl 2 -wallet file:C:/wallets/server start
```

Note: There is an `-ssl 4` option, whereby the SSL certificates are used only for mutual authentication but no encryption of data is done. This is an option provided for customers who have deployed all the applications behind a firewall and are not concerned about unauthorized interception of the data exchanged between the Fusion Client and the RPAS Server.

The syntax for starting up DomainDaemon is the same; replace `-ssl 2` with `-ssl 4`.

For additional information on SSL options, refer to SSL Options.

Setting Up a Client Key Store

Use the following instructions to set up a client key store.

1. Create an Oracle Wallet with Auto Login using this format: `orapki wallet create -wallet {client_wallet_directory} –auto_login -pwd {client_wallet_password}`

Example 5–12 Oracle Wallet with Auto Login

```
orapki wallet create -wallet C:/wallets/client –auto_login -pwd clientpass1
```

2. Generate a private key using this format: `orapki wallet add -wallet {client_wallet_directory} -keysize {key_len} -dn {client_dn} -pwd {client_wallet_password}`

Note: Oracle OSSA requires that the minimum key size is 2048.

Example 5–13 Private Key

```
orapki wallet add -wallet C:/wallets/client -keysize 2048 -dn "cn=rpas_qa_
client,dc=us,dc=oracle,dc=com" -pwd clientpass1
```

3. Export a certificate request using this format: `orapki wallet export -wallet {client_wallet_directory} -dn {client_dn} -request {client_cert_req_file} -pwd {client_wallet_password}`

Example 5–14 Certificate Request

```
orapki wallet export -wallet C:/wallets/client -dn "cn=rpas_qa_
client,dc=us,dc=oracle,dc=com" -request C:/wallets/client_cert_req.txt -pwd
clientpass1
```

4. Create the client certificate.

Note: For additional information, refer to the section, "Creating a Self-signed Root Certificate".

   a. If a third-party Certificate Authority (CA) is used, send the client certificate request file (`{client_cert_req_file}`) to the CA. The CA sends back a certificate for the client (referred to as `{client_certificate_file}`) along with the public certificate of the CA (referred to as `{root_cert_chain_file}`).

   b. If the self-signed root certificate is used, run the following command:

```
orapki cert create -wallet {root_wallet_directory} -request
{client_cert_req_file} -cert {client_certificate_file} -validity
{validity_days} -pwd {root_wallet_password}
```

Example 5–15 Self-signed Root Certificate

```
orapki cert create -wallet C:/wallets/root -request C:/wallets/client_cert_req.txt
-cert C:/wallets/client_cert.txt -validity 3650 -pwd rootpass1
```

5. Import the CA or self-signed certificate into the wallet using this format: `orapki wallet add -wallet {client_wallet_directory} -trusted_cert -cert {root_cert_chain_file} -pwd {client_wallet_password}`

Example 5–16 Import Certificate

```
orapki wallet add -wallet C:/wallets/client -trusted_cert -cert C:/wallets/root_
chain.txt -pwd clientpass1
```

> **Note:** For third-party CA, the public certificate chain may contain more than one certificate. These certificates need to be imported one by one starting from the top of the chain.

6. Import the client certificate into the wallet using this format: `orapki wallet add -wallet {client_wallet_directory} -user_cert -cert {client_certificate_file} -pwd {client_wallet_password}`

Example 5–17 Import the Client Certificate

```
orapki wallet add -wallet C:/wallets/client -user_cert -cert C:/wallets/client_
cert.txt -pwd clientpass1
```

Setting Up a Trust Store in the JKS Format

A trust store in the JKS format must be created. Then the CA or self-signed root certificate of the server certificate should be imported into the trust store.

Use the following instructions to set up a a trust store in the JKS format.

1. Import the CA or self-signed certificate for the server certificate into the wallet using this format: `orapki wallet add -wallet {client_wallet_directory} -trusted_cert -cert {root_cert_chain_file_for_the_server} -pwd {client_wallet_password}`

> **Note:** This step can be skipped if both client and server certificates are issued by the same CA or generated from the same self-signed root certificate.

Example 5–18 Import the CA or Self-signed Certificate

```
orapki wallet add -wallet C:/wallets/client -trusted_cert -cert C:/wallets/root_
chain.txt -pwd clientpass1
```

2. Convert the Oracle Wallet to JKS format using this format: `orapki wallet pkcs12_to_jks -wallet {client_wallet_directory} -pwd {client_wallet_password} -jksKeyStoreLoc {key_store_file} -jksKeyStorepwd {key_store_`

```
password} -jksTrustStoreLoc {trust_store_file} -jksKeyStorepwd {trust_
store_password}
```

Note: The password, `jksKeyStorepwd`, must be the same as what was entered into the WebLogic domain credential store using the credential entry alias *fckey*.

Example 5–19 Convert the Oracle Wallet to JKS Format

```
orapki wallet pkcs12_to_jks -wallet C:/wallets/client -pwd clientpass1
-jksKeyStoreLoc testkey.jks -jksKeyStorepwd welcome1 -jksTrustStoreLoc
testtrust.jks -jksTrustStorepwd welcome1
```

3. Finally, copy the files `testkey.jks` and `testtrust.jks` as `fckey.jks` and `fctrust.jks` into the **idstores** directory under the Fusion Client installation.

Note: The `fckey.jks` password is the client keystore password as requested in Figure 5–11, "Application Server Details Window".

Now the client key store setup is complete.

OBIEE Report Configuration Tasks

If the BI module is installed, proceed with the OBIEE report configuration tasks and setup as documented in the *Oracle Retail Predictive Application Server Administration Guide for the Fusion Client*

Clear the Browser Cache

After the Fusion Client has been upgraded, ensure that all users clear their browser cache.

Review the RPAS Configuration Property Files

If you install the RPAS Fusion Client over an existing RPAS Fusion Client environment, the installer updates the existing `config/config.properties` and `config/rpas/rpasConfig.properties` files. If you modified any property values in these files, they are overwritten by the files found in the installer. However, the installer backs up the existing files and creates a report on the changed property values and any new properties added that did not exist in the previous version.

Example 5–20 RPAS Configuration Property Files

Following is an example of the `config.properties`.

```
-rw-r--r--  1 build users  375 Jun  7 17:18 config.properties
-rw-r--r--  1 build users  401 Jun  7 16:44 config.properties.201106071642
-rw-r--r--  1 build users  258 Jun  7 16:44
config.properties.201106071642.mergeReport
```

This example has the following components:

- `config.properties` is the new file
- `config.properties.201106071642` is the backup of the previous version of config.properties

- `config.properties.201106071642.mergeReport` contains what was added, updated, or removed between `config.properties.201106071642` and `config.properties`.

You can review the changes and restore any property value if needed.

Set Up Single Sign-On (SSO)

RPAS can be set up on an Oracle SSO infrastructure that enables users who are already connected and authenticated to the Oracle SSO to directly access the RPAS Fusion Client or RPAS Classic Client Web Deployment. For more information about SSO, refer to "Appendix: Oracle Single Sign-On (SSO)".

To set up SSO, perform the following steps:

1. Set up the Identity Management Infrastructure for SSO. Install the following components:
   - Oracle Identity Management (OID) LDAP server. For more information, refer to the *Oracle Fusion Middleware Installation Guide for Oracle Identity Management 11g Release 1 (11.1.1)*.

     Note: To avoid authentication issues with SSO users, select the **Use Retrieved User Name as Principal** check box and then restart your WebLogic server. This option is available in the OID provider on the **Provider Specific** tab in the **Users** section.

2. An Oracle Internet Directory repository configured to be used by the LDAP server. (OID uses an Oracle Database as the back end.) Ensure that you have a WebLogic domain extended with the JRF template. This was already done before you installed the RPAS Fusion Client. For more information, refer to Setting Up the WebLogic Server.
3. Skip this step if you are setting up SSO for RPAS Classic Client Web Deployment.

   During the RPAS Fusion Client installation, specify that you want to use the SSO feature to log on to the application and specify the SSO users or groups. For more information, refer to the Installation Properties File Parameter Reference section or see Steps 16, 23, and 24 in the section, Installing the RPAS Fusion Client in Graphical or Text Mode.
4. Install the Oracle Fusion Middleware 11g Web Tier Utilities referring to the *Oracle Fusion Middleware Installation Guide for Oracle Web Tier* guide.

If you are setting up an SSO environment, continue to Setting Up an SSO Environment.

Setting Up an SSO Environment

For additional information, refer to "Appendix: Oracle Single Sign-On (SSO)" and specifically, Figure D–1, "Single Sign-On Topology".

Perform the following steps if you are setting up an SSO environment:

1. Register the Oracle HTTP server (Web Tier) with the Oracle Access Manager server by configuring a Webgate plug-in. To do this, perform the following steps:
   a. From the Webgate installation, copy certain essential files over to the Web Tier installation. For more information, refer to the instructions at this address:

http://download.oracle.com/docs/cd/E14571_01/install.1111/e12002/webgate004.htm.

Note: Webgate needs operating system specific Gnu C Compiler (GCC) libraries to be available on the library load path. For Linux and Solaris you can obtain the library from http://gcc.gnu.org/. More details on this requirement can be found at the following address: http://docs.oracle.com/cd/E14571_01/install.1111/e12002/webgate002.htm#CACGIGBB.

b. Create a Webgate definition using the Oracle Access Manager console (http://<host-ip-addr>:<OAM-domain-port>/oamconsole). The Webgate's base URL must point to the Web Tier server. This creates files under **<OAM Domain dir>/output** directory (`cwallet.sso` and `ObAccessClient.xml`).

c. Copy those files from the web tier to: `Oracle_WT1/instances/instance1/config/OHS/ohs1/webgate/config`.

2. From the Oracle SSO server administrator, obtain the OID information, such as the TCP/IP address and port, whether SSL is used as a transport mechanism, and the realm name. In addition, obtain an administrative login and password, such as the one used by the `orcladmin` user.

3. Configure authentication providers in your WebLogic domain for SSO, by performing the following steps:

a. If not already present, then copy `oamAuthnProvider.jar` from the **$MW_HOME/oracle_common/modules/oracle.oamprovider_11.1.1** directory to the **/u00/webadmin/Oracle/Middleware/wlserver_10.3/server/lib/mbeantypes** directory.

b. Configure the OAM identity asserter using the instructions at the following address: http://docs.oracle.com/cd/E14571_01/core.1111/e10043/osso.htm#CHDCDAIC. Specifically, refer to the section "9.3.5.2.4 - Setting up providers for identity assertion."

c. Configure an OID authenticator so that all the roles are retrieved from OID for the user logged in through SSO. Following are the key parameters you need to enter in the provider-specific configuration tab:

Parameter	Enter:
Group Search Base	cn=Groups, dc=us,dc=*yrcmpny*,dc=com
User Search Base	cn=Users, dc=us,dc=*yrcmpny*,dc=com
User name attribute	uid
location	ipaddress:portnumber, for example 10.10.10.10:2345
bind dn	cn=<LDAP admin user name>
password	<LDAP admin user's password>

Note: Where *yrcmpny* is your company's name.

d. Ensure the three asserters are configured in the following order:

- OAM Identity Asserter: OPTIONAL

- OID Authenticator: SUFFICIENT
- DefaultAuthenticator: SUFFICIENT

e. Save and activate the changes.

f. Restart the WebLogic server.

4. Update the `mod_wl_ohs.conf` file to send requests to the WebLogic server. To update the `mod_wl_ohs.conf` file, perform the following steps:

   a. Navigate to the location where the `mod_wl_ohs_conf` file exists and open it for editing. For example,

      ```
      $ORACLE_INSTANCE/ config/ <COMPONENT_TYPE>/ <COMPONENT_NAME>
      ```

   b. Update the file based on Example 5–21 and Example 5–22.

Example 5–21 Single WebLogic Instance

For a single WebLogic instance, specify:

```
<Location /console>
   SetHandler weblogic-handler
   WebLogicHost server1
   WebLogicPort 7001
</Location>
```

This forwards `/console` from the HTTP server to `/console` on the WebLogic Server with the host name and port number, server1:7001.

Example 5–22 WebLogic Instances in a Cluster

For WebLogic instances in a cluster, specify:

```
<Location /myServerURL>
   SetHandler weblogic-handler
   WebLogicCluster server1:7010,server2:7010
   WLCookieName RPASJSESSIONID
</Location>
```

This forwards `/myServerURL` from the HTTP server to `/myServerURL` on the WebLogic Clusters with the host names and port numbers, server1:7010 and server2:7010.

Note: In Example 5–21 and Example 5–22, server1 and server 2 are the host names used for illustrative purposes. Ensure that you use relevant host names, port numbers, and context roots based on your implementation.

Oracle Linux on x86 Architecture

There is a known issue with the JDK running on an Oracle Linux server with Intel x86 processors. For the steps to avoid this error, refer to StringIndexOutOfBoundsException in Oracle Linux on x86 Architecture.

Upload Application Security Policies to OID-based Domain Policy Store

Before using the Fusion Client, users and groups need to be authorized to access the application. If using a file-based Oracle Fusion Middleware policy store (the default when you create a WebLogic domain) the Fusion Client installer takes care of this

automatically. The users and groups specified in the installer properties are automatically provided with application access.

If you are using Oracle Internet Directory (OID) as your policy store, then the installer does not set up user and group authorization; you must follow these steps to complete this aspect of the installation.

For guidance on setting up an OID-based policy store, refer to the chapter "Configuring the OPSS Security Store" in the Oracle® Fusion Middleware Application Security Guide 11g Release 1 (11.1.1) Configuring the OPSS Security Store. This document is available through My Oracle Support.

Procedure

Upload Application Security Policies to OID-based Domain Policy Store using the following steps:

> **Note:** Before you begin, ensure that the RPAS Fusion Client is installed and the OID policy store is already set up.

1. Verify your RPAS Fusion Client application name and version. This can be found on the Deployments page in the WebLogic Application Console,

   For example: RPAS Fusion Client application name is `RPASFusionClient1` and version is `14.0.0.444.201309030533.256083`.

2. Determine your application policy stripe. The application policy stripe is your RPAS Fusion Client application name plus # plus version.

   For example: Application policy stripe is `RPASFusionClient1#14.0.0.444.201309030533.256083`

3. Extract META-INF/jazn-data.xml from `rpas.ear` (located under <cd-image-dir>/application) to a local directory..

4. Change directories to: `cd  <this local dir>/META-INF`.

5. Copy `<WLDOMAIN>/config/fmwconfig/jps-config.xml` to your local directory.

6. Copy the domain bootstrap credential wallet to your local directory. `cp <WLDOMAIN>/config/fmwconfig/bootstrap`

7. Add a service instance definition to jps-config.xml:

```
<serviceInstance name="atlantis.policystore.xml" provider=
"policystore.xml.provider">
         <property name="location" value="./jazn-data.xml"/>
         </serviceInstance>
```

8. Add two jps contexts:

```
<jpsContext name="ldap-policy-store">
               <serviceInstanceRef ref="policystore.ldap"/>
       </jpsContext>
       <jpsContext name="Atlantis">
               <serviceInstanceRef ref="atlantis.policystore.xml"/>
       </jpsContext
```

9. Launch the WLST shell and run the migrateSecurityStore command using your Application policy stripe for <policystripe>:

```
$ORACLE_HOME/oracle_common/common/bin/wlst.sh
migrateSecurityStore(type="appPolicies", configFile="./jps-config.xml",
src="Atlantis", dst="ldap-policy-store", srcApp="Atlantis",
dstApp="<policystripe>", overWrite="true", mode="lax")
```

10. Use the Enterprise Manager Fusion Middleware tool to locate the Fusion Client application security configuration, and add users and groups as members of the "atlantis-users" role.
11. Restart your application server.
12. Delete the META-INF directory since it is no longer needed.

Enable Gzip Compression

Enable Gzip compression to reduce network traffic.

Compressing the HTTP responses provides significant benefit to the Fusion Client. Using gzip compression compresses the HTTP response by a factor of 10 to 20 in most cases. This greatly reduces the available bandwidth consumed by each Fusion Client user, which leads to greater scalability in that a much larger number of concurrent users can be accommodated than would otherwise be possible.

The main reason for the large degree of compression is that the HTML content is inherently verbose, and in the case of the Workbook page there is a great deal of repetitive content in the HTML markup corresponding to the various worksheet cells.

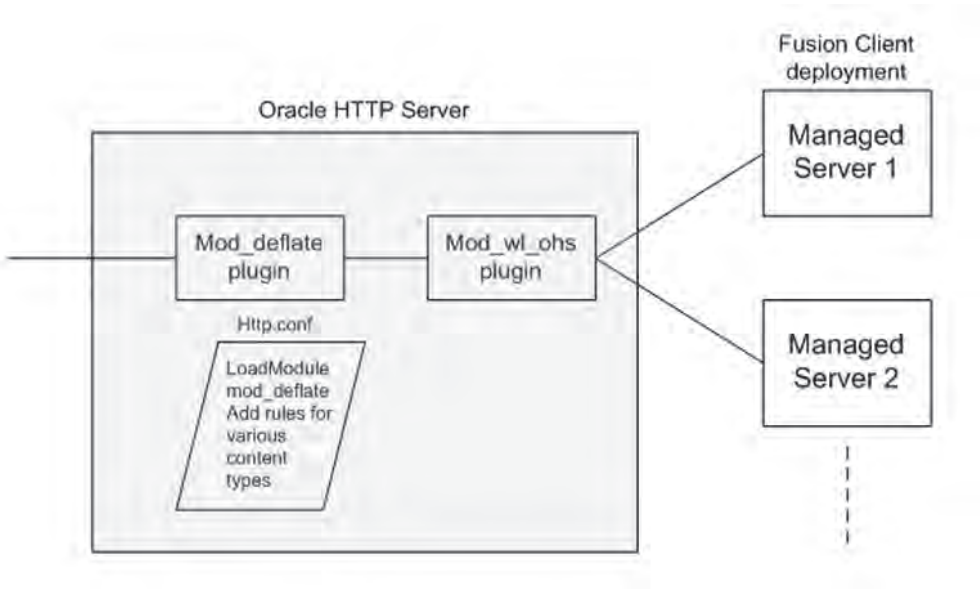
Another benefit is faster end user response time, though this is rather unpredictable; sometimes the improvement can be very small. This is because the size of the HTTP response has its greatest effect on the latency, the time it takes for all the response bytes to completely arrive at the client browser; and latency is just one component of the overall response time. Other factors like server processing time, time to render the response on the browser, time for the request to reach the server, are also important, as well as, the need to account for the time taken to uncompress the response.

The decompression time can vary from browser to browser.

Implementing Compression

While there are many ways to achieve response compression, it has been tested using Oracle's web tier technologies. A basic web tier consists of an Oracle HTTP server instance furnished with a mod_wl_ohs module plug-in that tunnels user requests to WebLogic managed servers hosting Fusion Client application instances. Implement gzip compression using the mod_deflate module in the Oracle HTTP Server as shown in Figure 5–25.

Figure 5–25 Implementing Gzip Compression



The configuration of the mod_deflate module is to be done in the httpd.conf file corresponding to the Oracle HTTP Server instance, by adding the following lines:

```
LoadModule deflate_module "${ORACLE_HOME}/ohs/modules/mod_deflate.so"
AddOutputFilterByType DEFLATE text/plain
AddOutputFilterByType DEFLATE text/xml
AddOutputFilterByType DEFLATE application/xhtml+xml
AddOutputFilterByType DEFLATE text/css
AddOutputFilterByType DEFLATE application/xml
AddOutputFilterByType DEFLATE image/svg+xml
AddOutputFilterByType DEFLATE application/rss+xml
AddOutputFilterByType DEFLATE application/atom_xml
AddOutputFilterByType DEFLATE application/x-javascript
AddOutputFilterByType DEFLATE text/html

SetOutputFilter DEFLATE
```

You can verify that gzip compression is being applied by looking at the response headers using a tool such as Firebug (on Mozilla Firefox); you should see the following line in the headers.:

```
Content-Encoding gzip
```

Note: For further details, refer to the My Oracle Support guide, "How to Add Compression on the Oracle HTTP Server to Reduce the Download Size of Webcenter Spaces Deployment ()" (Doc ID 1219013.1).

Troubleshooting

This section lists possible solutions for some issues that may occur when using the application.

WebLogic Ulimit Error

An error may occur when starting the WebLogic server if the limit for open files is set too low:

```
<Log Management> <BEA-170019> <The server log file /u01/Oracle/Middleware/user_
projects/domains/domain1/servers/AdminServer/logs/AdminServer.log is opened. All
server side log events will be written to this file.>
Unhandled exception
Type=Segmentation error vmState=0x00040000
J9Generic_Signal_Number=00000004 Signal_Number=0000000b Error_Value=00000000
Signal_Code=00000033
Handler1=09001000A049F450 Handler2=09001000A0495F70
R0=0000000000000001 R1=000000011757E200 R2=09001000A0340048 R3=09001000A0340178
R4=0000000000000234 R5=0000000000000000 R6=800000000000F032 R7=3000000000000000
R8=000000000012F7D0 R9=0000000000000000 R10=0000000000000000 R11=F0000000301D3600
R12=090000000154ABB0 R13=0000000117588800 R14=0000000116635CA0
R15=0000000116599D00
```

To avoid this issue, set the ulimit to a larger number, at least 8192, by issuing the following command:

```
ulimit -u 8192
```

Error Occurs When Users Access the Charting Feature in the RPAS Fusion Client

If users attempt to access the application from a system that does not have a connected Display/Video card, they may encounter the following error message when they try using the charting feature for the first time:

```
Sun.awt.X11GraphicsEnvironment (initialization failure)
For more information, please see the server's error log for an entry beginning
with: Server Exception during PPR, #
```

To avoid this issue, edit the WebLogic Startup script with the java.awt.headless parameter using the following steps:

1. Navigate to the following location on the system where the application server is installed:

   ```
   <WEBLOGIC_DOMAIN_HOME>/bin/
   ```

2. In a text editor, open the `startWebLogic.sh` script for editing.

3. Append the following parameters to the set `JAVA_OPTIONS = %SAVE_JAVA_OPTIONS%` statement:

   ```
   -Djava.awt.headless=true
   ```

   After it is set up, the JAVA_OPTIONS statement appears as in the following example:

   ```
   set JAVA_OPTIONS=%SAVE_JAVA_OPTIONS% -Djava.awt.headless=true
   ```

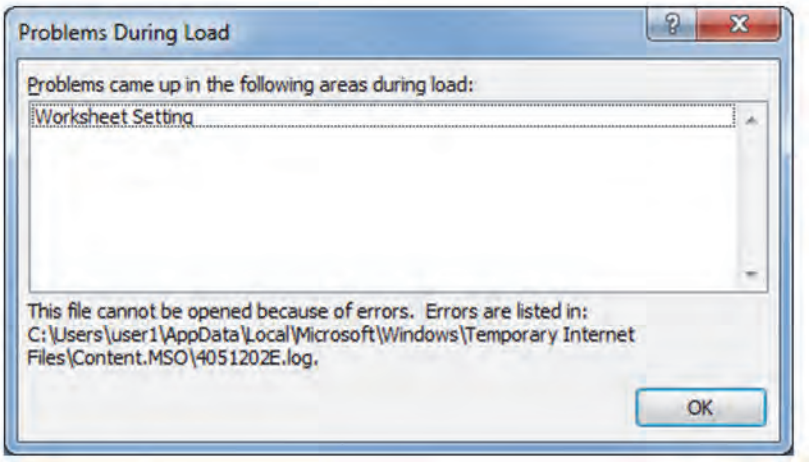
4. Save and close the file.

5. For the changes to take effect, restart the WebLogic Server.

Error Occurs When Users Export to Microsoft Excel in the RPAS Fusion Client

If you attempt to export to Microsoft Excel in the Fusion Client, you may encounter the error message shown in Figure 5–26

Figure 5–26 Microsoft Excel Error Message



This error message means that the Excel export file is corrupt due to a headless environment. In the server logs, the error message appears as follows:

```
ERROR view - java.lang.NullPointerException
java.lang.NullPointerException
        at javax.swing.MultiUIDefaults.getUIError(MultiUIDefaults.java:133)
        at javax.swing.UIDefaults.getUI(UIDefaults.java:741)
        at javax.swing.UIManager.getUI(UIManager.java:1002)
        at javax.swing.JPanel.updateUI(JPanel.java:109)
        at javax.swing.JPanel.<init>(JPanel.java:69)
        at javax.swing.JPanel.<init>(JPanel.java:92)
        at javax.swing.JPanel.<init>(JPanel.java:100)
```

To avoid this issue, edit the WebLogic Startup script with the `java.awt.headless` parameter using the following steps:

1. Navigate to the following location on the system where the application server is installed:

   `<WEBLOGIC_DOMAIN_HOME>/bin/`

2. In a text editor, open the `startWebLogic.sh` script for editing.

3. Append the following parameters to the set `JAVA_OPTIONS = %SAVE_JAVA_OPTIONS%` statement:

   `-Djava.awt.headless=true`

   After it is set up, the JAVA_OPTIONS statement appears as the following example:

   `set JAVA_OPTIONS=%SAVE_JAVA_OPTIONS% -Djava.awt.headless=true`

4. Save and close the file.

5. For the changes to take effect, restart the WebLogic Server.

StringIndexOutOfBoundsException in Oracle Linux on x86 Architecture

If you attempt to access the application from a system that does not have a connected Display/Video card, you may encounter the following error message when you try using the charting feature for the first time:

StringIndexOutOfBoundsException in org.apache.myfaces.trinidadinternal.style.util.CSSGenerationUtils

To avoid this issue, you must set up the WebLogic Startup script with the JVM option -XX:-UseSSE42Intrinsics parameter. To do this, perform the following steps:

1. Navigate to the following location on the system where the application server is installed:

   `<WEBLOGIC_DOMAIN_HOME>/bin/`

2. In a Text Editor, open the `startWebLogic.sh` script for editing.
3. Append the following parameters to the set `JAVA_OPTIONS = $SAVE_JAVA_OPTIONS` statement:

   `-XX:-UseSSE42Intrinsics`

   After set up, the JAVA_OPTIONS statement appears like the following example:

   ```
   set JAVA_OPTIONS=$SAVE_JAVA_OPTIONS -XX:-UseSSE42Intrinsics
   ```

4. Save and close the file.
5. Remove the cached CSS file, for example:

   ```
   $APP_TMP_
   FOLDER/public/adf/styles/cache/blafplus-desktop-ezog8j-en-ltr-gecko-1.9
   .1.8-cmp.css
   ```

6. For the changes to take effect, restart the WebLogic Server.
7. Clear the browser cache.

6

Installing and Configuring the RPAS Classic Client

This section describes the installation of the RPAS Classic Client on Windows machines, and describes how to configure the client to connect to a domain.

Make RPAS Classic Client Files Generally Accessible

Perform the following procedure to make the RPAS Classic Client available.

1. Create a directory on the network from where users will install the RPAS Classic Client.

   The location and the name of the directory are up to the system administrator's preferences. This directory is henceforth referred to as the **[RPASCLIENT]** directory.

2. Copy the files from the following directory on the server:

   [RPAS Installation]/Client

   to the **[RPASCLIENT]** directory.

Installing the RPAS Classic Client

The RPAS Classic Client installation procedure is the same for all of the RPAS applications. Perform the following procedures to install the application on a PC.

1. Run the `setup.exe` file located in the **[RPASCLIENT]** directory on the network.
2. The welcome page displays. Follow the installation procedures as prompted.

   The setup program exits after the installation is complete.

Configuration

After creating an RPAS domain and starting the DomainDaemon (refer to the *Oracle Retail Predictive Application Server Administration Guide for the Classic Client*), you must configure the RPAS Classic Client to connect to the domain on a server. The eConfigure utility is used to create this connection.

For information about eConfigure, refer to the eConfigure section in the *Oracle Retail Predictive Application Server Administration Guide for the Classic Client*.

7

RPAS Classic Client Web Deployment

The RPAS Classic Client can be deployed through traditional installation or through Web-based environments. This chapter describes the RPAS Web deployment installation process.

> **Note:** Accessing RPAS through a Web-based deployment is different from the RPAS Fusion Client. For more information on the Fusion Client, refer to Chapter 5, "Installing the RPAS Fusion Client".

Web-based deployment allows you to perform the following:

- Use a Web browser to install the RPAS Classic Client application to the user's computer.
- Launch the RPAS Classic Client when it has already been installed.
- Reinstall the RPAS Classic Client when an updated version is available.
- Use the RPAS Web Launch applet to facilitate In-Context Launch integration.

For the supported components used for web deployment of the Classic Client, refer to Table 1–2, " RPAS Classic Client Hardware and Software Requirements".

These instructions assume that the previously specified software has been properly installed and configured. Consult the documentation of each component for installation and configuration information, as well as hardware and software requirements.

For the RPAS Web deployment to function properly, users must have sufficient access to their PCs (typically administrator rights) which allow them to install software, unless the administrator configures the applet to launch only preinstalled RPAS Classic Client. Specifically, they need permission to write into the Windows Registry.

Installation and Configuration Process Overview

The following is an overview of the process that must be followed to install RPAS for Web deployment.

- Install the RPAS Web Application. This installation is completed onto the Web server and involves two components that are included with the RPAS archive (`RPAS.war` or `RPAS_osso.war`, and `RPASWebData.tar`).
- Install multiple versions of RPAS Classic Client files on Web server (as needed).
- Configure the RPAS Servlet by using the deployment descriptor `web.xml` to specify servlet properties.

- Configure Oracle SSO for RPAS Web application (if Oracle SSO is used).
- Start the RPAS Web Configuration Utility. Using the URL of the RPAS Web Launch application, administrators and users follow this process to log in to the system.
- Perform other Web client administration activities. Once the Web deployment environment is prepared, additional configuration and administration activities, such as domain configuration and managing administrative users, may need to be performed.

Installing the RPAS Web Application

Installing the RPAS Web Application consists of the following procedures:

- Preparing Your Environment
- Installing the necessary files and configuring the environment based on your type of installation. Different processes may be used for RPAS Web deployment:
  - Installing on WebLogic Server with SSO Support
  - Installing on WebLogic Server without SSO Support
  - Installing on Apache Tomcat (a standalone server that is not part of the SSO infrastructure)
- Configuring the RPAS Servlet
- Configuring and Administering the Web Application

Preparing Your Environment

Perform the following procedure to prepare your environment to install t he RPAS web application.

1. Log in to the UNIX server and determine where the RPAS Web files will be installed. A minimum of 50 MB disk space available is required for the application installation files. More space may be needed if multiple versions of RPAS Classic Client are supported on the Web server.
2. Copy the RPAS Web files (`RPAS.war` or `RPAS_osso.war`, and `RPASWebData.tar`), located in **[RPAS Installation]/Web/** directory, to a newly created staging directory on the UNIX server. This directory will be referred to as **STAGING_DIR**.
3. Extract the RPASWebData.tar to the appropriate location. This location is referred to as **[RPAS_WEB_DATA_DIR]** in this document. If the Web server is running in a load balance environment with multiple servers, the RPASWebData files must be deployed to a network drive accessible to all Web server instances. A new directory **RPASWebData/** and three subdirectories (**client/**, **db/**, and **logs/**) are created. Verify that the client directory has read permissions and that the db and logs directories have read and write permissions.
4. For each release of RPAS Classic Client, there are two files: `buildNumber.txt` and `client.zip`. These files are not part of RPAS Web files. They generally come with RPAS release package. The default installation location for the files is **[RPAS_WEB_DATA_DIR]/RPASWebData/client**. If multiple client versions are to be supported, both files of each version must be placed under **[RPAS_WEB_DATA_DIR]/RPASWebData/client/[VERSION]** where **[VERSION]** is the version number of that release (for example, 13.3, 12.0.10).
5. Perform the necessary procedures based on your type of implementation.

Installing on WebLogic Server with SSO Support

Perform the following procedure if you are implementing RPAS Web on a WebLogic server with SSO support. This process consists of these steps:

- Step 1: Set Up SSO
- Step 2: Configure the Oracle Access Manager
- Step 3: Set Up the WAR File
- Step 4: Deploy the WAR File
- Step 5: Set RPAS Role for Oracle SSO Logins

Step 1: Set Up SSO

This SSO configuration is the same as it is for SSO for the RPAS Fusion Client. Follow the procedure as described in the section, Set Up Single Sign-On (SSO). Ensure that a location directive for the RPAS Web context root has been added to the `mod_wl_ohs_conf` file. The context root is defined in Step 4: Deploy the WAR File.

Step 2: Configure the Oracle Access Manager

Configure the Oracle Access Manager using the following steps:

1. Log in to the OAM console.
2. Under Policy configuration in the left pane, select Shared Components > Authentication Schemes > Select LDAP Scheme.

   The LDAP Scheme window opens in the right pane.

   In the Challenge Parameter field, enter: *ssoCookie=disablehttponly*.

Step 3: Set Up the WAR File

Perform the following procedure to set up and deploy the WAR file to the WebLogic Server.

1. Configure the Application for the OSSO Identity Asserter.

   The WebLogic Server supports adding multiple authentication-methods. If you are setting up an OSSO Identity Asserter in the WebLogic Application Console, the Web application using the OSSO Identity Asserter must have its auth-method set to CLIENT-CERT. After deploying the application on the WebLogic Server, all `web.xml` files in the application EAR file must include CLIENT-CERT in the element auth-method for the appropriate realm. To edit `web.xml` for the OSSO Identity Asserter:

   a. Locate the web.xml file in the application WAR file. For example:

      ```
      WEB-INF/web.xml
      ```

   b. Locate the auth-method for the appropriate realm and enter CLIENT-CERT. For example:

      ```
      <login-config>
        <auth-method>CLIENT-CERT</auth-method>
        <realm-name>myrealm</realm-name>
      </login-config>
      ```

   c. Save the file.

d. Create a new `weblogic.xml` file with the following contents (replace [CONTEXT_ROOT] with the actual context root):

```
<?xml version='1.0' encoding='UTF-8'?>
<weblogic-web-app xmlns="http://xmlns.oracle.com/weblogic/weblogic-web-app"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://xmlns.oracle.com/weblogic/weblogic-web-app
http://xmlns.oracle.com/weblogic/weblogic-web-app/1.0/weblogic-web-app.xsd"
>

<context-root>[CONTEXT_ROOT]</context-root>

<security-role-assignment>
     <role-name>RPAS_ADMIN_ROLE</role-name>
     <principal-name>RPAS_ADMIN_ROLE</principal-name>
</security-role-assignment>

<security-role-assignment>
     <role-name>RPAS_USER_ROLE</role-name>
     <principal-name>RPAS_USER_ROLE</principal-name>
</security-role-assignment>

</weblogic-web-app>
```

e. Save the `weblogic.xml` file to the same location of the `web.xml` file in the WAR file.

2. **Configure the RPAS Web Launch** - To configure RPAS Web Launch, you need to modify one property file (`propfile`) located within the **WEB-INF/config** directory of the `RPAS_osso.war` file.

a. Locate the following information in `propfile` and replace [RPAS_WEB_DATA_DIR] with the actual location and [HOSTNAME] with the host name of the server)

```
dbPath=[RPAS_WEB_DATA_DIR]/RPASWebData/db
clientSourceDir=[RPAS_WEB_DATA_DIR]/RPASWebData/client
webLogFile=[RPAS_WEB_DATA_DIR]/RPASWebData/logs/rpasPortal.[HOSTNAME].log
isOSSO=true
debug=false
classicMode=false
launchPreinstalledOnly=false
supportMultipleVersions=true
```

Note: The `isOSSO` flag must be set to **true**. Set `launchPreinstalledOnly` to **true** if only pre-installed RPAS Classic Client can be launched. Set `classicMode` to **true** to support RPAS Classic Client release 9.4. To support multiple versions of RPAS Classic Client, set `supportMultipleVersions` to **true**.

---

b. Save the file and the WAR archive.

Step 4: Deploy the WAR File

Perform the following procedure to deploy the WAR file to the WebLogic Server:

1. Log on to the WebLogic Administration Console.
2. Under the Domain Structure (left navigation pane), select Deployments. The Summary of Deployments window opens.

3. On the Summary of Deployments window, click Install. The Install Application Assistant window opens.
4. On the Install Application Assistant window, navigate to the location where you extracted the `RPAS_osso.war` (`[STAGING_DIR]/RPAS_osso.war`), select the file, and click **Next**.
5. Click the Install this deployment as an application option and then click **Next**. The next window displays optional settings. You can choose to set them up or accept the default values and proceed.
6. Optional. Enter relevant information for the optional settings.
7. Click **Finish**. The WAR file is deployed and it appears listed in the Summary of Deployments window.
8. On the Summary of Deployments window, click the deployment you added. The Settings for <deployment-name> window opens.
9. On the Configuration tab, under General, enter a relevant value in the **Context Root** field. You can choose to type RPAS or any other context root. This name is referred to as [CONTEXT_ROOT] in this document.
10. Click **Save**. The Save Deployment Plan Assistant window opens.
11. Enter or select a location for the deployment plan, and click **OK**.
12. Restart your deployment for the changes to take effect.

Step 5: Set RPAS Role for Oracle SSO Logins

There are two types of roles for RPAS Web Launch users: RPAS_ADMIN_ROLE and RPAS_USER_ROLE.

Both roles can launch the RPAS Classic Client and connect to a domain. Only RPAS_ADMIN_ROLE has the privilege to access the ADMIN interface.

It is recommended that RPAS_USER_ROLE be assigned to most OSSO users (such as planner) and RPAS_ADMIN_ROLE be assigned for a few power users (such as executive). This needs to be performed on the LDAP server storing the OSSO user information.

The roles can be created manually by using the OID DAS application. The oidadmin application or LDIF scripts may also be used to create users and roles. For more details, refer to the OID documentation.

Creating a Group Using the DAS Application

Use the following procedure to use the DAS application to create the RPAS_USER_ROLE and RPAS_ADMIN_ROLE accounts.

1. Access the DAS application.

   The DAS application is found in the following location:

   `http://<host>:<port>/oiddas`

   where `<host>` and `<port>` are the infrastructure or Oracle Identity Management OAS.

   Example: `http://hostname.example.com:7778/oiddas`
2. Click **login** and log in as `orcladmin` or another privileged user.
3. On the right-side of the page, select the **Directory** tab, and on the left side, select the **Groups** link.

4. Click **Create**.
5. Enter the name of the group to create (for example, RPAS_ADMIN_ROLE), the Display name, and a description.
6. Make sure the Group Visibility option is set to **Public**.
7. If you would like, add additional users. Scroll to the Members section and click **Add User** to add users to this group. You can also nest other groups as well. Members can be added at a later time as needed.
8. When all members have been added, click **Submit**.

Creating Groups from an LDIF Script

Alternatively, you can create the groups using an LDIF script. A template is shown in Example 7–1. Note that the following token `@BASE_REALM_DN@` needs to be replaced with installation specific value of the Realm Distinguished Name. Also, this script creates the group with a single member, `orcladmin`, as part of the group. Additional members may be added with more `uniquemember` attributes. You can run the script with the `ldapadd` command supplied with the Oracle Identity Management infrastructure OAS server.

Example 7–1 LDIF Script

```
# The LDIF template for creating RPAS_ADMIN_ROLE and RPAS_USER_ROLE groups in OID.
# RPAS_USER_ROLE
dn: cn=RPAS_USER_ROLE,cn=groups,@BASE_REALM_DN@
objectclass: top
objectclass: groupOfUniqueNames
objectclass: orclGroup
cn: RPAS_USER_ROLE
displayname: RPAS user role
description: RPAS user role
orclisvisible: true
owner: cn=orcladmin,cn=users,@BASE_REALM_DN@
uniquemember: cn=orcladmin,cn=users,@BASE_REALM_DN@

# RPAS_ADMIN_ROLE
dn: cn=RPAS_ADMIN_ROLE,cn=groups,@BASE_REALM_DN@
objectclass: top
objectclass: groupOfUniqueNames
objectclass: orclGroup
cn: RPAS_ADMIN_ROLE
displayname: RPAS Administrator role
description: RPAS Administrator role
orclisvisible: true
owner: cn=orcladmin,cn=users,@BASE_REALM_DN@
uniquemember: cn=orcladmin,cn=users,@BASE_REALM_DN@
```

Once the RPAS groups have been created with the LDIF script, you could use the OID DAS application to add more members to them.

For more information about SSO, refer to "Appendix: Oracle Single Sign-On (SSO)".

Installing on WebLogic Server without SSO Support

Perform the following procedure if you are implementing RPAS Web on a WebLogic Server instance without SSO Support. This process consists of several steps:

- Step 1: Configuring RPAS Web Launch

Step 1: Configuring RPAS Web Launch

To configure RPAS Web Launch, you need to modify one property file (`propfile`) located within the **WEB-INF/config** directory of the `RPAS.war` file.

1. Locate the following information in **propfile** and replace `[RPAS_WEB_DATA_DIR]` with the actual location and `[HOSTNAME]` with the host name of the server).

```
dbPath=[RPAS_WEB_DATA_DIR]/RPASWebData/db
clientSourceDir=[RPAS_WEB_DATA_DIR]/RPASWebData/client
webLogFile=[RPAS_WEB_DATA_DIR]/RPASWebData/logs/rpasPortal.[HOSTNAME].log
isOSSO=false
debug=false
classicMode=false
launchPreinstalledOnly=false
supportMultipleVersions=true
```

> **Note:** The `isOSSO` flag must be set to **false**. Set `launchPreinstalledOnly` to **true** if only pre-installed RPAS Classic Client can be launched. Set `classicMode` to **true** to support RPAS Classic Client release 9.4. Set `supportMultipleVersions` to **true** to support multiple versions of RPAS Classic Client.

2. Save the file and WAR archive.

Step 2: Deploying the WAR File

Perform the following procedure to deploy the WAR file to the Oracle Application Server without SSO.

1. Log on to the WebLogic Administration Console.
2. Under the Domain Structure (left navigation pane), select **Deployments**. The Summary of Deployments window opens.
3. On the Summary of Deployments window, click **Install**. The Install Application Assistant window opens.
4. On the Install Application Assistant window, navigate to the location where you extracted the `RPAS.war` (`[STAGING_DIR]/RPAS.war`), select the file, and click **Next**.
5. Click the Install this deployment as an application option and then click **Next**. The next window displays optional settings. You can choose to set them up or accept the default values and proceed.
6. Optional. Enter relevant information for the optional settings.
7. Click **Finish**. The WAR file is deployed and it appears listed in the Summary of Deployments window.
8. On the Summary of Deployments window, click the deployment you added. The Settings for <deployment-name> window opens.
9. On the Configuration tab, under General, enter a relevant value in the **Context Root** field. You can choose to type RPAS or any other context root. This name is referred to as [CONTEXT_ROOT] in this document.
10. Click Save. The Save Deployment Plan Assistant window opens.
11. Enter or select a location for the deployment plan, and click **OK**.

12. Restart your deployment for the changes to take effect.

Installing on Apache Tomcat

Perform the following procedure if you are deploying RPAS Web launch on a standalone Apache Tomcat server, which implies a non-SSO environment.

Installing RPAS Web Launch on Apache Tomcat consists of two steps:

- Step 1: Deploying the RPAS WAR File
- Step 2: Configuring RPAS Web Launch on Apache Tomcat

Step 1: Deploying the RPAS WAR File

Refer to your Apache Tomcat documentation for more details of how to deploy a Web archive.

1. Log on to the Tomcat Web server URL: `http://server:port/manager/html`.
2. Scroll to **Deploy directory or WAR file** located on server section.
3. In the Context Path field, type **/RPAS** (or anything else you choose).

   This location is referred to (without the "/") as [CONTEXT_ROOT] in this document.
4. In the WAR or Directory URL field, type `[STAGING_DIR]/RPAS.war`.
5. Click **Deploy**.

The display name should show RPAS Web Launch for a successful deployment. In the event you need to a re-deploy RPAS Web Launch, it is recommended that the following be performed:

- Undeploy the Web application.
- Restart the Web server to clear any caching.
- Re-deploy the Web application.

Step 2: Configuring RPAS Web Launch on Apache Tomcat

To configure RPAS Web Launch, you need to modify one property file (`propfile`). This file is located in the following path; where `[TOMCAT_INSTALL_DIR]` is the installation location of the Tomcat Web server. If the Web server is running in a load balance environment with multiple servers, this file must be modified for all Web server instances:

`[TOMCAT_INSTALL_DIR]/webapps/[CONTEXT_ROOT]/WEB-INF/config`

1. Locate the following information in `propfile` and replace `[RPAS_WEB_DATA_DIR]` with the actual location and `[HOSTNAME]` with the host name of the server.

```
dbPath=[RPAS_WEB_DATA_DIR]/RPASWebData/db
clientSourceDir=[RPAS_WEB_DATA_DIR]/RPASWebData/client
webLogFile=[RPAS_WEB_DATA_DIR]/RPASWebData/logs/rpasPortal.[HOSTNAME].log
isOSSO=false
debug=false
classicMode=false
launchPreinstalledOnly=false
supportMultipleVersions=true
```

Note: The `isOSSO` flag must be set to **false**. Set `launchPreinstalledOnly` to **true** if only pre-installed RPAS Classic Client can be launched. Set `classicMode` to **true** to support RPAS Classic Client release 9.4. Set `supportMultipleVersions` to **true** to support multiple versions of RPAS Classic Client.

2. Restart the RPAS Web Launch application.

Migrating from Previous Versions

If you have been running an older version of RPAS Web Launch, here are the steps to migrate old data to the new deployment.

1. Migrate client binary:

   Copy `buildNumber.txt` and `client.zip` files to the location specified by property `clientSourceDir`. If multiple versions are supported, they should be copied to `clientSourceDir/`[VERSION] where [VERSION] is the version number of that release (12.1.2, 11.1.15, and so on).

2. Migrate admin user data:

   Copy `userdata.dat` file to the location specified by property `dbPath`.

3. Migrate domain registration data:

   Copy `domaindata.dat` to the location specified by property `dbPath`.

4. Restart Web application RPAS Web Launch.

Note: After the migration, if an admin user fails to log on, that would indicate that the `userdata.dat` file is corrupt. Remove the file, and log on the administration interface using default user, **adm** (default password **adm**), and re-create all admin users.

Configuring the RPAS Servlet

The class for the RPAS servlet is com.retek.mdap.servlet.ServletManager. The servlet properties have been configured in the deployment descriptor web.xml that is originally archived in `RPAS_osso.war` or `RPAS.war`.

This deployment descriptor provides two sets of initialization parameters to the RPAS servlet.

Note: The deployment descriptor should not be modified.

After the servlet is configured, load it into your Web server. You might be required to reload your Web server to activate the new servlet.

Configuring and Administering the Web Application

The following topics provide information on accessing the RPAS Web Administration console to perform administrative tasks such as defining the RPAS enterprise configuration, and adding, modifying and deleting domain configurations.

Start the RPAS Web Configuration Utility – Administration Console

Perform the following procedure to start the RPAS Web Configuration utility.

1. To access the RPAS Web configuration utility, start a Web browser and go to the following location; where `WEB_SERVER_ADDRESS` is the address you use to access your Web server and`CONTEXT_NAME` is the value you defined for the Context Root field as described in each of the three installation processes listed in the Installing the RPAS Web Application section:

   `http://[WEB_SERVER_ADDRESS]/[CONTEXT-NAME]/web`

   Note: The Web application support internationalization. It uses the locale from the browser to determine the appropriate language to display in the Web interface. The default language is English.

   Example: `http://rpasweb.oracle.com:13085/RPAS/web`

   Depending the type of RPAS deployment being implemented (with SSO or without SSO), one of the following windows opens.

Figure 7–1 Login Window after OSSO Authentication

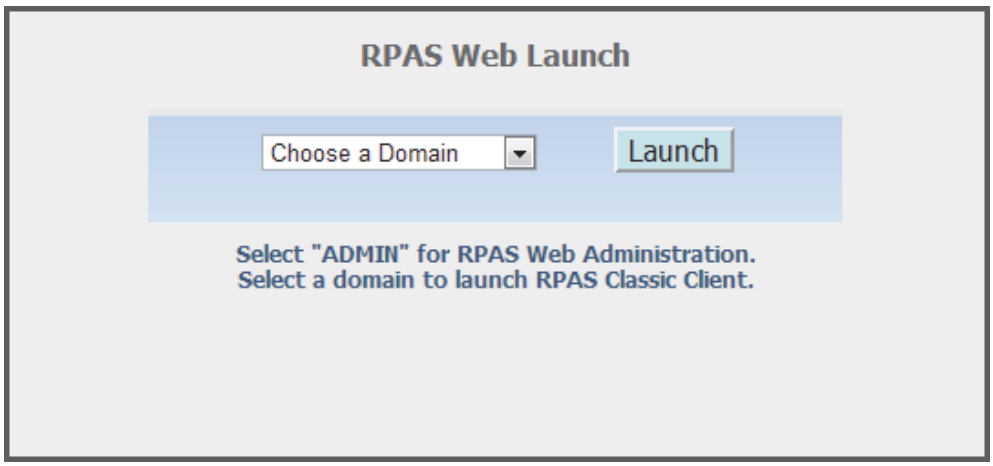
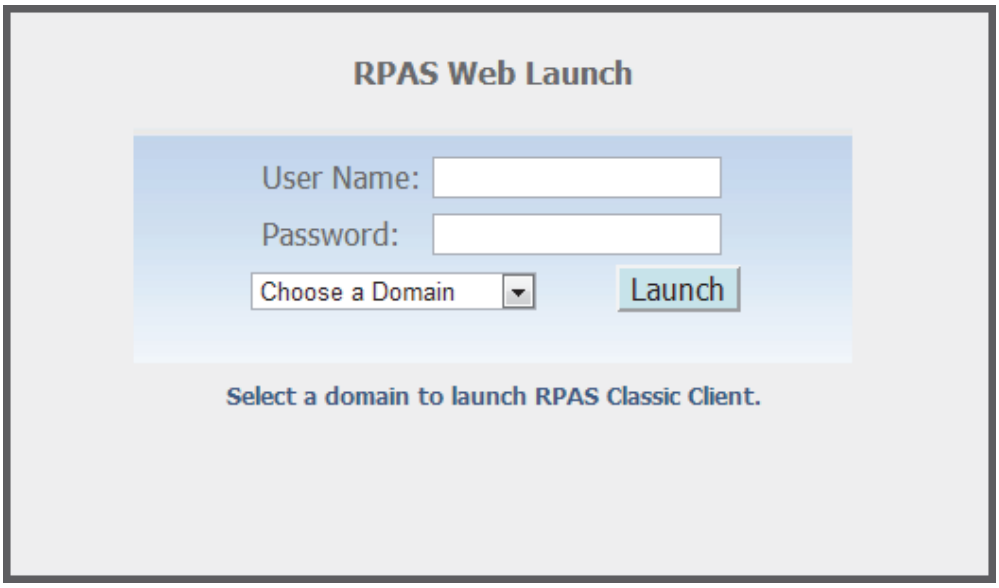


Figure 7–2 Login Window for non-OSSO Configuration

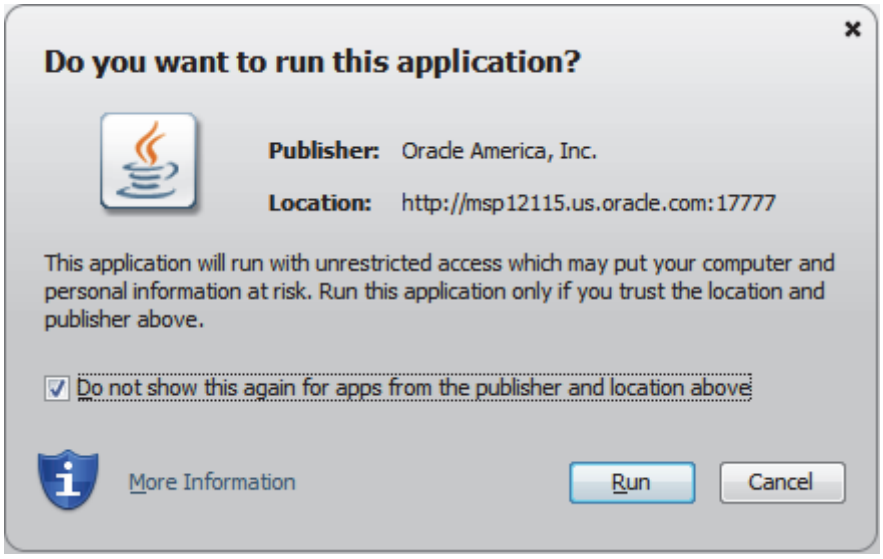


Note: If there is a very long list of domains, use URL `http://[WEB_SERVER_ADDRESS]/[CONTEXT-NAME]/web?app=[AppID]` to filter domains on the login page. Only domains with an application ID field matching `AppID` will be displayed in the list.

2. Perform one of the following:
   - If you are using an SSO environment, select **ADMIN** as the domain and click **Login** to access the Administration Console.
   - If you are not using an SSO environment, enter an Administrator user name and password (the initial administration user name is **adm** and the password is **adm**). Select **ADMIN** as the domain and click **Login** to access the Administration Console.

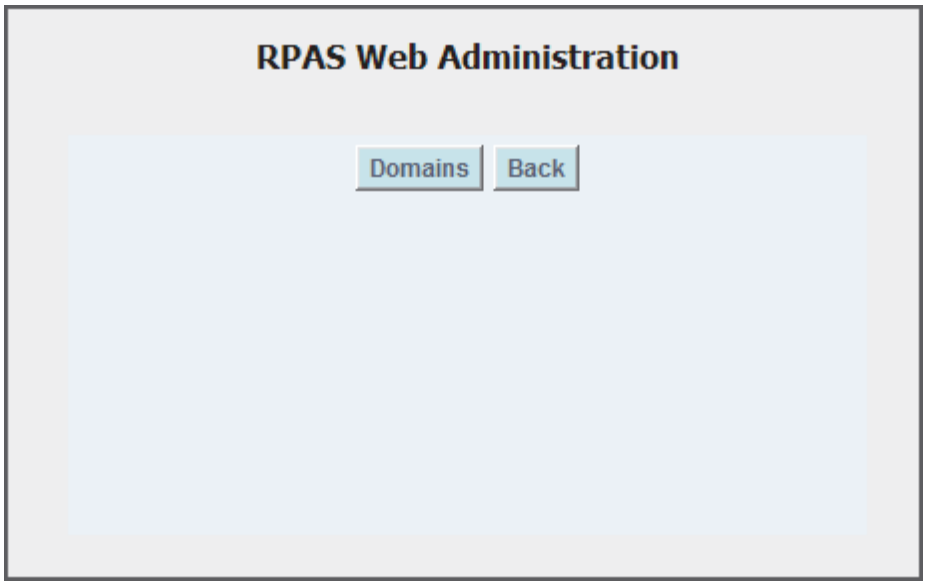
   A security warning opens.

Figure 7–3 Security Warning on Internet Explorer



3. Click **Run**. To avoid seeing this message in the future, make sure **Always trust content from this publisher** option is selected. The RPAS Web Administration console opens.

Figure 7–4 RPAS Web Administration Console in SSO Environment



4. Refer to the following topics to configure RPAS Web Launch or perform other administration activities.

Other Web Client Administration Activities

The following sections provide instructions for these other Web Client Administration activities:

- Adding, Modifying and Deleting Domain Configuration in an SSO Environment

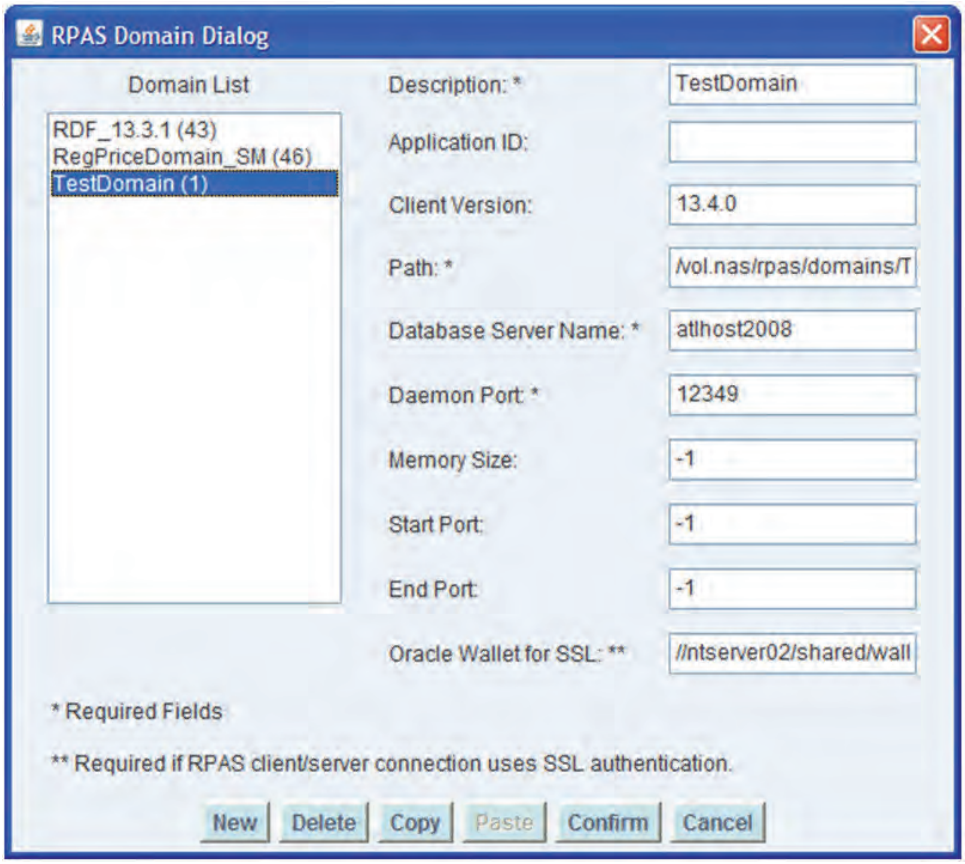
- Adding, Modifying and Deleting Domain Configuration in a Non-OSSO Environment

Adding, Modifying and Deleting Domain Configuration in an SSO Environment

Perform the following procedure to add, modify, and delete domain configuration in an SSO environment:

1. Click **Domains** in the RPAS Web Administration Console. The RPAS Domain Dialog window opens. Use this window to specify the location of RPAS domains. Each domain that can be accessed by a user must be specified with the dialog.

Figure 7–5 RPAS Domain Dialog



   a. To add a new domain, click **New**, enter the following information, and click **Confirm**.

Field Name	Value Description	Use
Description	This is displayed to users when they are selecting a domain to log in to.	Required.
Application ID	Used in domain filtering.	Can be any string without spaces. Leave blank if preferred.
Client Version	The version number of the RPAS Classic Client to launch.	It must match exactly the version number in the path of the client files on the Web server. Leave blank if multiple version support is not enabled.

Field Name	Value Description	Use
Path	The full path to the directory containing the domain on the database server.	Required.
Database Server Name	The hostname of the database server containing the domain.	Required.
Daemon Port	The port number of the DomainDaemon process running on the database server. The port must be between 1025 and 65535 (inclusive).	Required.
Memory Size	Not currently used.	Leave blank.
Start Port	Start of the range of ports used by a client PC (Web launch architecture) to connect to the database server.	This value must be great than (>) 1025. If it not specified, the RPAS database server attempt to find a free port whenever a client connects.
End Port	End of the range of ports used by a client PC (Web launch architecture) to connect to the database server.	This value cannot be greater than 65535.
Oracle Wallet for SSL	The wallet location that passes to the Classic Client in order to support one-way SSL.	Required if the RPAS client/server connection uses SSL.

b. To change an existing domain configuration, select the domain from the **Domains List**, modify the fields as necessary, and click **Confirm**. Select **Cancel** to discard any changes that have been made.

c. To remove a domain, select a domain from the **Domains List** and click **Delete**. The selected domain configuration is removed.

d. To copy all of the domain settings of a domain, perform the following:

- Select the domain from the **Domains List** and click **Copy**.
- Selecting another domain from the **Domains List** and click **Paste**. The domain is updated the domains settings you have copied.
- Click **Confirm** to save the updated information.

Adding, Modifying and Deleting Domain Configuration in a Non-OSSO Environment

In a non-OSSO environment, the administration tasks of maintaining the Domain data file are done manually by the administrator on the web server. The domain data file, `domaidata.dat`, can be located by the property `dbPath` in the property file (propfile) under the WEB-INF/config directory of the deployment.

The `domaindata.dat` file uses a Java property file format, that is with key=value format. If the key or value fields has any special characters (like #, !, =, and :) in it, then these characters should be preceded with backslash to ensure that they are properly loaded. Spaces are not allowed at the end of each line.

The domain configuration key follows the format domain.{number_sequence}.{property_name}. The number sequence begins from 1. The number sequence can be discrete. But they should be unique for each domain configuration

Table 7–1 Domain Configuration Key Descriptions

Field Name	Property Name in File	Value Description	Use
Description	.description	This is displayed to users when they are selecting a domain to log in to.	Required.
Application ID	.applicationID	Used in domain filtering.	Can be any string without spaces. Leave blank if preferred.
Client Version	.clientVersion	The version number of the RPAS Classic Client to launch.	It must match exactly the version number in the path of the client files on the Web server. Leave blank if multiple version support is not enabled.
Path	.path	The full path to the directory containing the domain on the database server.	Required.
Database Server Name	.dbServerName	The hostname of the database server containing the domain.	Required.
Daemon Port	.daemonPort	The port number of the DomainDaemon process running on the database server. The port must be between 1025 and 65535 (inclusive).	Required.
Memory Size	.dbMemSize	Not currently used.	Leave blank.
Start Port	.dbPortStart	Start of the range of ports used by a client PC (Web launch architecture) or the Web server (Web tunneling architecture) to connect to the database server.	This value must be great than (>) 1025. If it not specified, the RPAS database server attempt to find a free port whenever a client connects.
End Port	.dbPortEnd	End of the range of ports used by a client PC (Web launch architecture) or the Web server (Web tunneling architecture) to connect to the database server.	This value cannot be greater than 65535.
Oracle Wallet for SSL	.sslWalletLocation	The wallet location that passes to the Classic Client in order to support one-way SSL.	Required if the RPAS client/server connection uses SSL.
id	.id	Should be same as the number_ sequence value for the current domain configuration.	Used for internal mapping.
count	.count	Total domain configuration count	Not used now. Used for backward compatibility

Variable: domaindata.dat file

The domain count entry in the domaindata.dat file is no longer used. However, this variable is allowed to be present in the domaindata.dat for backward compatibility.

This is a global variable, used for maintaining the total domain configuration count.

This follows the format domain.count.

Example 7–2 Sample domaindata.dat file

```
domain.1.sslWalletLocation=/path/to/wallets/client
domain.1.path=/domain/path/in/server
```

```
domain.1.daemonPort=12345
domain.1.clientVersion=14.0.0
domain.1.dbPortStart=-1
domain.1.dbServerName=serverName or IPAddress
domain.1.id=1
domain.1.dbMemSize=-1
domain.1.dbPortEnd=-1
domain.1.applicationID=
domain.1.description=TestDomain
domain.count=10
```

Install and Launch the RPAS Classic Client Application

Perform the following procedure the install the RPAS Classic Client and log in to a domain using RPAS Web Launch:

1. Start a Web browser and go to the following location/URL: `http://[WEB_SERVER_ADDRESS]/[CONTEXT_NAME]/web`

   Example: `http://rpasweb.oracle.com:13085/RPAS/web`

   This address is established during the initial installation and configuration. Administrators must provide this location/URL to end users. The `[WEB_SERVER_ADDRESS]` portion of the URL is the host address where the Java application service is running. This address may also include an alternate TCP/IP port number to communicate on (for instance, for port 8080, webss:8080). The login window opens.

   Note: If using Windows 7, start the web browser in *Run as administrator* mode the first time you use RPAS Web Launch.

   This ensures that the browser has privileges to install the RPAS Classic Client. After the first time, run the web browser in regular mode for better security.

   ---

2. Perform one of the following based on your environment:

   - If your environment is not using Oracle SSO, enter a user name and password, select a domain from the list, and then click **Login**.
   - If you are using SSO, enter your SSO credentials for authentication. A drop-down list appears, displaying a list of domains you can choose from. Select a domain. The RPAS Classic Client login page opens. Log in with your RPAS solution username and password.

     Note: When using SSO, you can by-pass the login page by specifying the domain in the URL:
     `http://[WEB_SERVER_ADDRESS]/[CONTEXT_NAME]/web?domain=[Desc].`

     The domain with a description field matching Desc will be launched automatically after the authentication. No spaces are allowed in the description field if this direct triggering mechanism feature is used.

When**Login** is selected, the **DomainDaemon** on the database server is contacted to verify that the specified user is allowed to access the selected domain. Ensure that the **DomainDaemon** process is running on the database server before clicking **Login**.

If access to the domain is allowed, a security dialog window may appear.

3. If the security window opens, click **Run**.

> **Note:** Java Security Settings should be set to Medium instead of High. From the Windows Control Panel, click the Java icon to open the Java Control Panel. Select the Security tab and change the Security Level to Medium and then click **OK**.

After you click **Run** in the security window, a check is made to see if the RPAS Classic Client application needs to be installed on the user's PC. The Web server administrator is able to define a common installation location of the RPAS Classic Client for all users' PCs.

This is accomplished by setting the appropriate value in the server-side `clientPath.txt` file (note the mixed-case filename) under the **clientSourceDir/[VERSION]** directory where the optional **[VERSION]** is the client version number if multiple versions are supported. The file `clientPath.txt` is an optional file which must reside under the same directory as `buildNumber.txt` does. RPAS Web installation files do not contain it. The administrator, or person responsible for installing RPAS Server components, must decide whether or not to define the installation location on user's PC by creating this file and specifying the full path of installation directory in the first line of the file. If **launchPreinstalledOnly** flag is set to **true**, the Web Launch applet will try to launch RPAS Classic Client from that location without downloading and installing the client. It will fail if the RPAS Classic Client has not been installed, and the applet will display an error message. If **launchPreinstalledOnly** flag is set to **false** and the user has not previously installed the RPAS Classic Client, or a newer version has become available on the server, the RPAS Classic Client will be downloaded and installed. If the **launchPreinstalledOnly** flag is set to **false** and the RPAS Classic Client path is not specified, the user is prompted for an installation location for the RPAS Classic Client. The RPAS Classic Client installation directory must have at least 50 MB storage space.

4. If necessary, select a directory that has at least 50 MB of free storage for installing RPAS Classic Client, and click **OK**. A status dialog box opens as files are copied from the server to the user's PC. After the files have been copied, a RPAS installation program runs, and the RPAS Classic Client starts. If everything is successful, the user sees a **Login Successful** message in the bottom left corner of the RPAS Classic Client window.

> **Note:** If the RPAS Classic Client does not need to be installed on the user's PC after you click **Login**, the RPAS Classic Client immediately starts and connects the user to the selected domain.

Troubleshooting

If a problem is encountered when using RPAS Web Launch, review any log files and record the output to determine the causes. Any support ticket submitted to Oracle must have the logging output attached.

On the server side, the log files are specified by the property `webLogFile`. On the client side, logging output is written to Java Console which can be opened by right-clicking the Java icon in the Windows system tray and choosing **Open Java Console**.

If the browser fails to launch the client without displaying an error message or behaves abnormally, we recommend that the user clear all browser cookies and try again.

If an instance of RPAS Classic Client is already running when the Web Launch applet is trying to install RPAS Classic Client, the installation may fail. We recommend that the user stop all RPAS Classic Client processes and try again.

Note: You can turn on the `debug` flag to obtain additional logging information by setting `property debug` to **true** in the property file `propfile` and restarting the web application.

RPAS Web Launch and Web-based Applications

If you plan to implement RPAS Web Launch (including In-Context Launch) in conjunction with web-based applications, refer to either the Classic Client or Fusion Client version of the *Oracle Retail Predictive Application Server Administration Guide* for more information.

8

Installing and Building the RPAS HSA Environment

This chapter describes the full installation process of an Hybrid Storage Architecture (HSA) integration system on RPAS. This chapter includes these topics:

- RPAS HSA Prerequisites
- RPAS HSA Build Process
- RDM Repository
- RDM Schema Information
- Integration Configuration
- Partition Information
- RDM SSL Configuration
- RPAS Installation Utilities

> **Note:** For information on upgrading or patching HSA, see HSA Patch Installation.

RPAS HSA Prerequisites

These prerequisites must be met before building the integrated environment, RPAS HSA:

- Oracle Database Server

  The Database server must have an Oracle Database 12c installed. The database connection details, that is, the server's IP address, port number, and SID or instance's service name should be provided to the RPAS installer. So the installer can use the connection information for the HSA installation.

- Oracle Database Client

  If the Oracle database server is not the same as the RPAS server, the Oracle Database 12c Client needs to be installed on the RPAS Server, including SQL Plus and SQL Loader. And ORACLE_HOME must also be set.

- Participating Domains

  Before registering each domain, it must:

  - Be upgraded to RPAS version 14.1.or later
  - Conform to the RPAS Data Mart (RDM) hierarchy structure

- Be pre-built and loaded with hierarchy data
- Have all informal positions formalized if removed before registration. DPM operations can resume after registration.
- Be identified by the `checkRDM` utility as a valid domain.

Note: For additional information about RDM, refer to RDM Repository.

RPAS HSA Build Process

The RPAS HSA Build process has three major steps:

1. Preinstallation
2. Installation
3. Postinstallation

Preinstallation

Unlike any other traditional installer, database schemas are created as part of the RPAS HSA installation process.

The preinstallation process prepares the RDM repository, the Domains, Oracle wallet, and SQL script to create database schemas and roles. This SQL script must be manually run by a Database administrator.

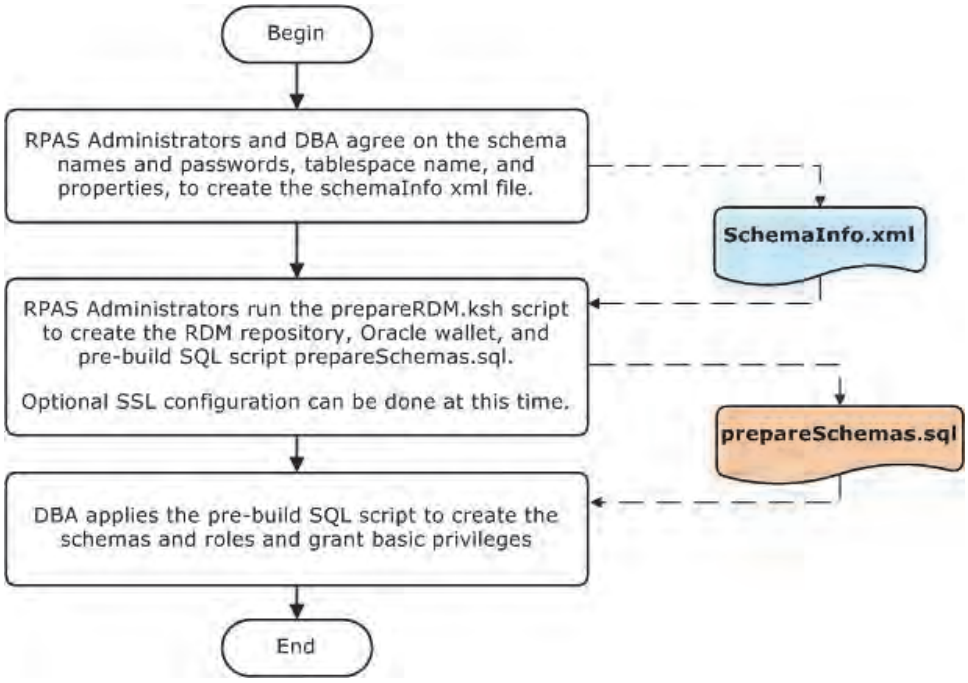
Prepare the RDM Repository

Figure 8–1 shows the preinstallation process flow and domain preparation. Since the HSA integration is a pluggable feature of RPAS, ensure that the existing processes of building and patching individual domains has not changed. All other schemas are granted additional required privileges during the installation

In order to integrate the domains, they must be upgraded and patched to RPAS version 14.1 or later.

The RDM Data Mart Schema should be granted the privileges using the provided script. All other schemas should be granted the corresponding role. At this point, the roles have privilege to create private synonyms only.

Figure 8–1 Preinstallation: Prepare RDM Repository



Prepare the Domains

Since the HSA integration is an optional feature of RPAS, existing processes of building and patching individual domains remain unchanged. Before a domain is registered with an HSA environment, it must be upgraded to the current RPAS version and patched with any Domain Configuration changes.

Informal positions created through DPM are unique to the individual domain and not shared by other domains or RDM. In order to prevent naming conflicts, all informal positions on the shared dimensions must be either formalized or removed before domain registration with the RDM. After a domain has been registered, DPM operations can continue as before since the RDM contains a mechanism to prevent naming conflicts.

Installation

The installation process builds schema objects in the database schemas and grants privileges to each role.

The RDM is built and populated from integration configuration and partition information by the `rpasInstall` utility. The access schemas which are created in the preinstallation step are populated by the schema objects and metadata.

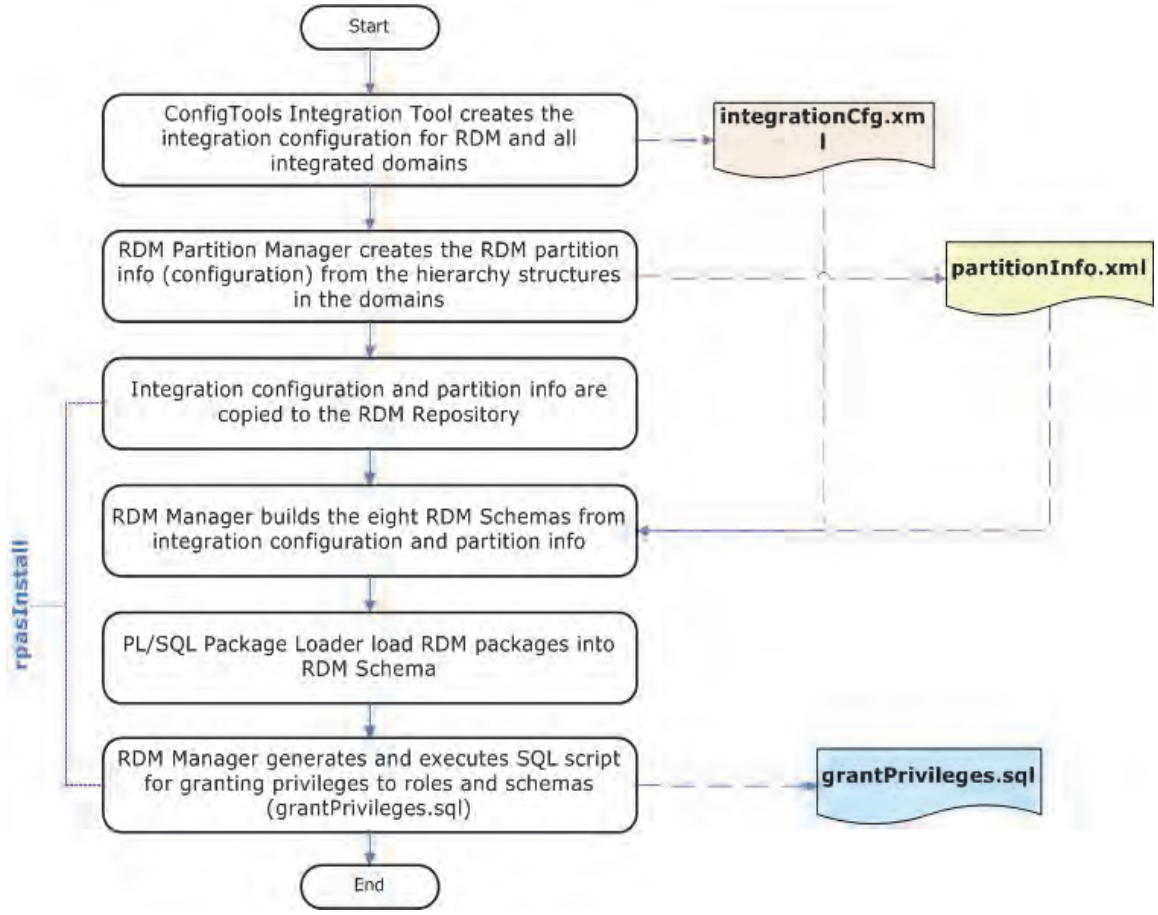
Figure 8–2 shows the process of building the RDM schemas. The RDM manager invoked by rpasInstall can directly build the schemas in the database, or use an Alternative Installation Process.

During this build process:

- All schema objects along with the PL/SQL packages are created in the data mart schema.
- Private synonyms pointing to the data mart schema objects are created in each user schema.

- All required privileges are granted to the roles.

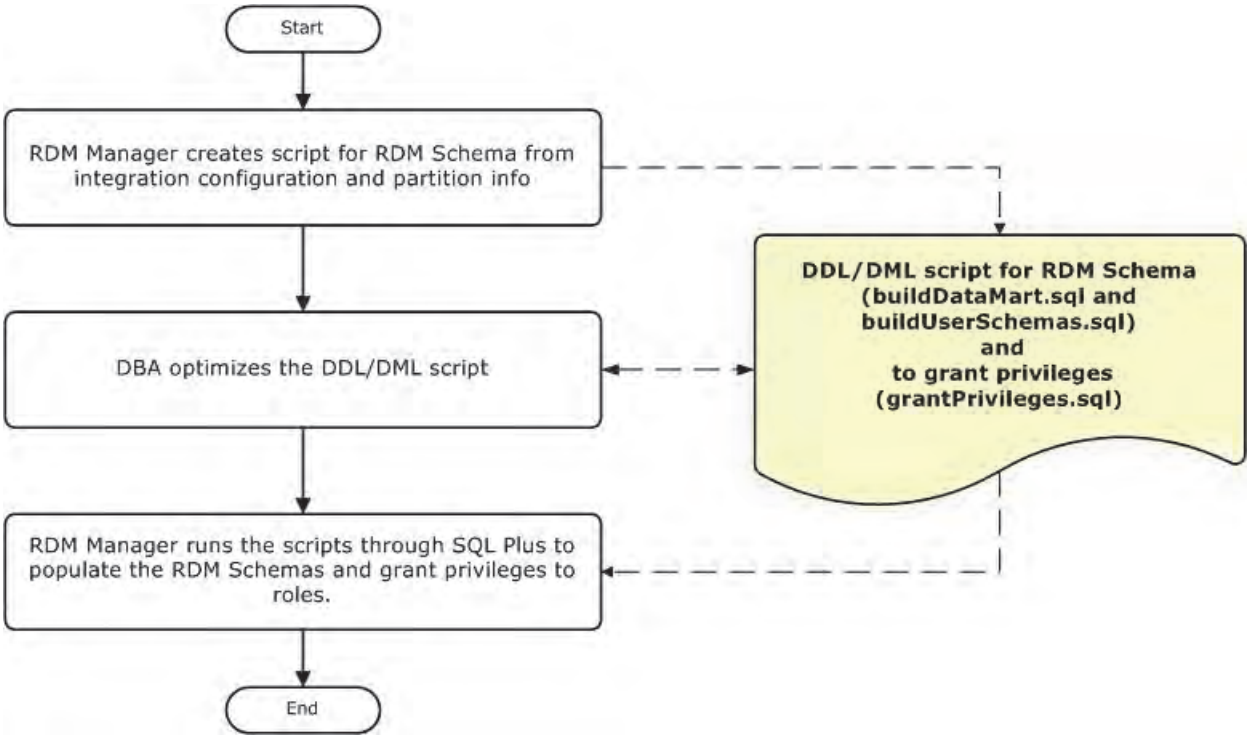
Figure 8–2 Installation: Building the RDM Schemas



Alternative Installation Process

An alternative process of building the RDM schema is shown in Figure 8–3. Two DDL/DML SQL script can be generated first, one for the data mart schema and one for other access schemas. After optimization by the DBA or Database performance expert, it can then be used to populate the schema.

Figure 8–3 Installation: Building the RDM Schemas - Alternative Process



Postinstallation

The postinstallation process initializes the data synchronization and registers compatible domains.

After the HSA system is built, the postinstallation process has three steps:

- Load Dimension Data
- Register Participating Domains
- Load Shared Fact Data

Figure 8–4 shows all of the post installation steps.

Load Dimension Data

The dimension data input files must first be copied to the **input/dimdata** directory in the RDM repository before running `loadDimData`. Perform this either manually or through a wrapper script.

Register Participating Domains

A domain can be registered anytime after the RDM has been built as long as it has been included in the integration configuration (domain name in the data flow set and shared measures in the shared fact set). This allows incremental integration of multiple domains.

If a domain has not been defined in the integration configuration, it requires a patching process to bring the domain into the HSA system.

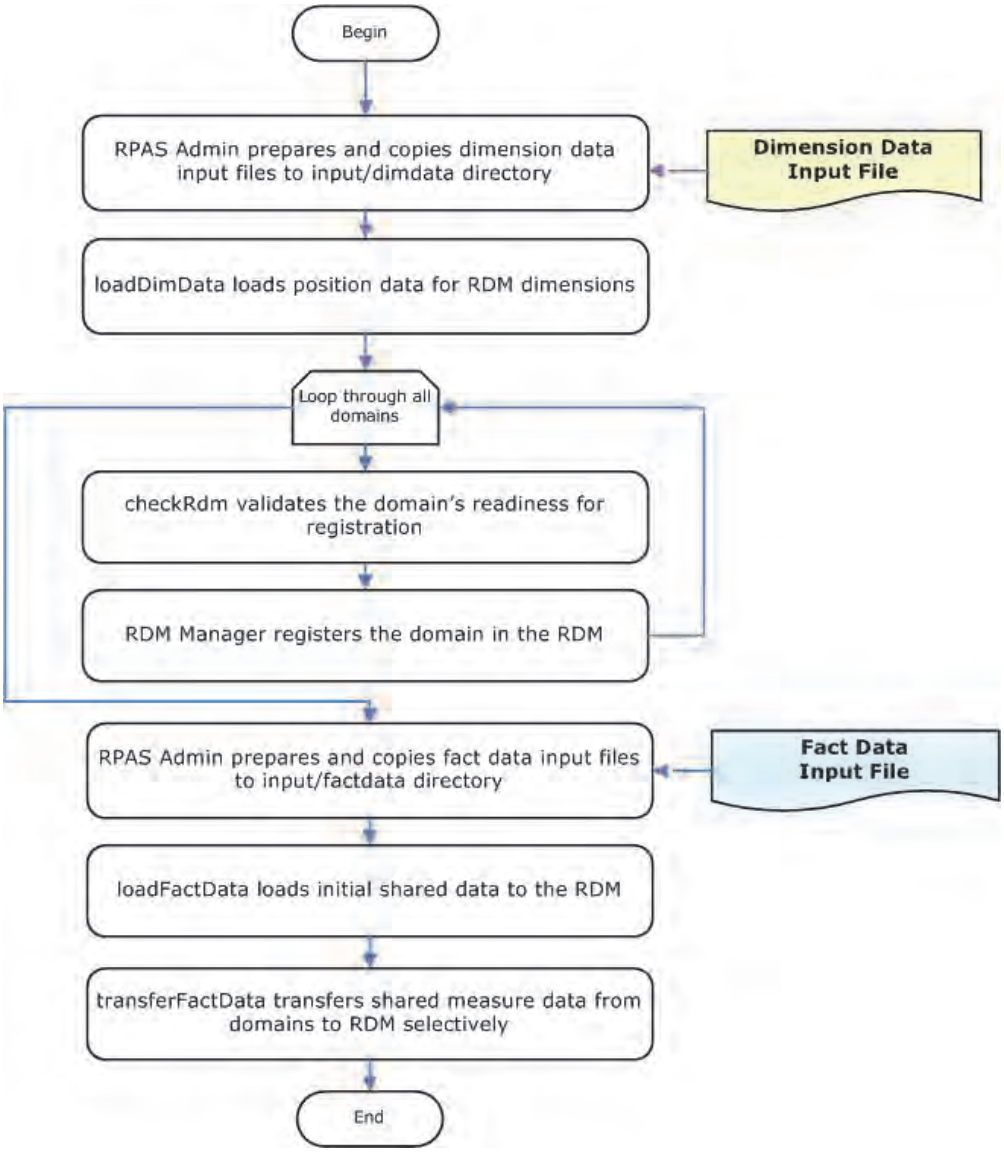
Only a valid domain can be registered. The validity of a domain is checked by its conformity with the RDM at the partition, position and ITT level. These checks are

performed by `rdmMgr` before registering. However, to check the complete validity of the domain, rollup conflicts must be verified. The `checkRDM` utility has the ability to verify a domain's validity.

Load Shared Fact Data

There are two ways to populate the shared data in RDM. The `loadFactData` utility can load fresh data from flat input files. The `transferFactData` utility can transfer existing data in the domains to RDM selectively according to user input.

Figure 8–4 Postinstallation: Initializing RDM Dimension, Fact Data, and Register Domains



RDM Repository

The RPAS Data Mart (RDM) repository is a directory on the file system to store RDM configuration, input files, log files, and metadata. Its directory structure is very similar to that of an RPAS domain except no dimension or measure data is stored there.

The RDM repository is created by `rdmMgr` when the RDM is built and updated when the RDM is patched as part of the preinstallation process. It is a centralized location for RDM properties, Oracle database credential wallets and network configuration files.

The location to create the RDM repository is specified when running the prepareRDM script.

RDM Repository Directory Structure

Figure 8–5 shows the directory structure of the RDM repository that is described in Table 8–1.

Figure 8–5 RDM Repository Directory Structure

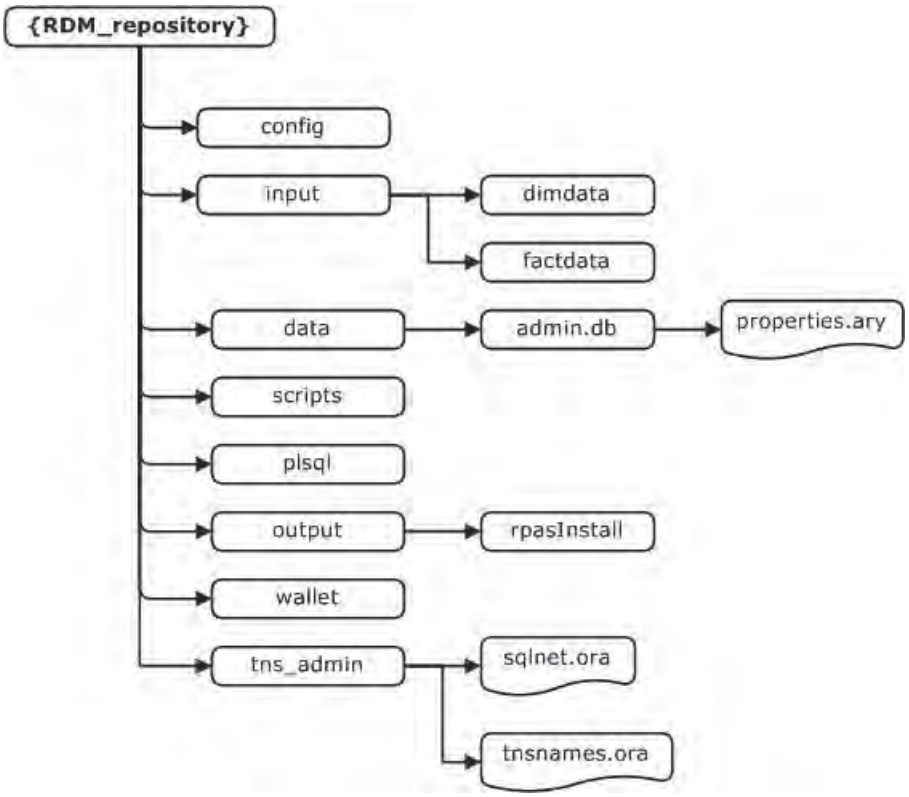


Table 8–1 RDM Repository Directory Structure Description

Directory	Contents	Populated By	Processed By
config	Contains schema information file and Integration and partition configuration files.	The `rdmMgr` copies the Schema information file from the user specified location to this location with a fixed name, after creating the RDM repository. The `rpasInstall` copies the configuration files from user specified locations to this location with fixed names.	The `rdmMgr` uses the files from this directory.
input	Contains shared hierarchies and measures data input files.	The input dimension input files should be manually copied to the **dimdata** folder and the fact data files copied to the **factdata** folder.	The `loadDimData` (using the **dimdata** subdirectory) and `loadFactData` (using **factdata** subdirectory) loads data from this directory and moves it to a processed folder.
data/admin.db	Contains `properties.ary` to store the data mart and user schema names.	rpasInstall/rdmMgr	Any operation connecting to the database.
scripts	Contains any script output of the RDM manager.	RDM Manager	DBA
plsql	Contains PL/SQL packages and procedures	`rpasInstall/buildSchema`	`rpasInstall/loadSqlPackages.ksh`
output	RDM log files	RDM utilities	
wallet	Oracle database credential and SSL wallet	`prepareRDM.ksh`	Any operation connecting to the database.
tns_admin	Oracle database network configuration (`tnsnames.ora` and `sqlnet.ora`)	RDM Manager	Any operation connecting to the database.

RDM Properties in a Domain

In an integrated domain, all RDM related properties are encapsulated in the array, `rdm_properties.ary`. Figure 8–6 shows the RDM-related entities in a domain that is described in Table 8–2.

Figure 8–6 RDM Properties in a Domain

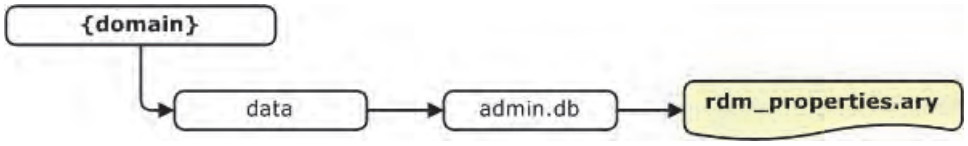


Table 8–2 RDM Properties in a Domain Descriptions

Directory	Contents	Populated By
data/admin.db	New array `rdm_properties.ary` stores `domain_name` and `rdm_path`.	`rpasInstall/rdmMgr`

RDM Schema Information

In order to implement RDM Schema Security, the DBA and the RPAS Administrator need to work together and set up a multi-schema environment. The schema names and connection parameters for the RDM are specified in a Schema Information file in xml format.

Table 8–3 lists all schemas and their corresponding roles. The Data Mart Schema is the home of all persistent data. Some of the Access Schemas require disk space for creating and populating staging tables during various RPAS operations. Tablespace quotas need to be granted accordingly.

Note: The recommended schema names can be changed, but the role names and database aliases are constants. The schema IDs must not be changed.

Table 8–3 RDM Schemas and Roles

Schema	Schema ID	Recommended Name	Role	DB Connect String (db_ alias)	Notes
RDM Data Mart Schema	rpas_data_mart	rpas_data_mart	N/A	rpas_data_mart_ conn	The main schema for all persistent data.
RDM Patch User	rpas_patch_user	rpas_patch_user	rpas_patch_role	rpas_patch_conn	
RPAS Batch User	rpas_batch_user	rpas_batch_user	rpas_batch_role	rpas_batch_conn	Requires space for fact staging tables.
RDM Dim Loader	rpas_dimload_ user	rpas_dimload_ user	rpas_dimload_ role	rpas_dimload_ conn	Requires space for dimension staging tables.
RDM Fact Loader	rpas_factload_ user	rpas_factload_ user	rpas_factload_ role	rpas_factload_ conn	Requires space for fact staging tables.
RPAS Hier Manager	rpas_hiermgr_ user	rpas_hiermgr_ user	rpas_hiermgr_ role	rpas_hiermgr_ conn	Requires space for DPM related staging tables.
RPAS Workbook User	rpas_wkbk_user	rpas_wkbk_user	rpas_wkbk_role	rpas_wkbk_conn	Requires space for staging tables for multiple users.
RPAS ETL Schema	rpas_etl_user	rpas_etl_user	rpas_etl_role	rpas_etl_conn	Requires space for fact staging tables

Schema Info Format

The schema names and Oracle database network configuration can be configured in the schemaInfo.xml file which is passed to the RDM schemas preparation script.

The tablespace properties, name and `space_quota`, are optional. If no tablespace is provided, then the default tablespace of the database is used.

If no default tablespace is specified, then the schemas with unspecified tablespace are created in the system tablespace.

It is important to have proper tablespace management. If no `space_quota` is provided, then no table space quota is granted to the user.

The space quota can be decided based on the information in the Notes column from Table 8–3.

The `space_quota` must be accompanied by the tablespace name. However, the `prepareSchemas.sql` can be modified manually before running.

The `<service_name>` or `<sid>` must be present and only one can be present. This file has to be provided while creating the RDM repository as explained in RDM Repository.

Example 8–1 Format for schemaInfo xml

```
<?xml version="1.0" encoding="UTF-8" ?>
<rdm_configuration>
   <schema_set>
      <schema id="rpas_data_mart">
  <name>rpas_data_mart</name>
        <tablespace>
     <name></name>
     <space_quota></space_quota>
  </tablespace>
      </schema>
      <schema id="rpas_patch_user">
   <name>rpas_patch_user</name>
        <tablespace>
     <name></name>
     <space_quota></space_quota>
  </tablespace>
      </schema>
      <schema id="rpas_batch_user">
   <name>rpas_batch_user</name>
        <tablespace>
     <name></name>
     <space_quota></space_quota>
  </tablespace>
      </schema>
      <schema id="rpas_dimload_user">
   <name>rpas_dimload_user</name>
        <tablespace>
     <name></name>
     <space_quota></space_quota>
  </tablespace>
      </schema>
      <schema id="rpas_factload_user">
   <name>rpas_factload_user</name>
        <tablespace>
     <name></name>
     <space_quota></space_quota>
   </tablespace>
      </schema>
      <schema id="rpas_hiermgr_user">
   <name>rpas_hiermgr_user</name>
        <tablespace>
     <name></name>
     <space_quota></space_quota>
  </tablespace>
      </schema>
      <schema id="rpas_wkbk_user">
```

```
   <name>rpas_wkbk_user</name>
         <tablespace>
     <name></name>
     <space_quota></space_quota>
  </tablespace>
      </schema>
      <schema id="rpas_etl_user">
   <name>rpas_etl_user</name>
         <tablespace>
     <name></name>
     <space_quota></space_quota>
  </tablespace>
      </schema>
   </schema_set>
   <tns_parameters>
      <protocol>tcp</protocol>
      <host>burrg41002v.us.oracle.com</host>
      <port>1521</port>
      <server>dedicated</server>
      <service_name>rpasodb.us.oracle.com</service_name>
      <sid></sid>
   </tns_parameters>
</rdm_configuration>
```

Integration Configuration

The Integration Tool of RPAS ConfigTools creates the integration configuration from user inputs and one or more domain configurations. A name is specified for the integration configuration, which is saved as the name attribute of the highest level XML element. Example 8–2 shows an integration configuration with a name specified as `RDM`.

The language attribute specifies the language of all the labels in the integration configuration. It is used as the source language in the translation tables for those labels. It can be any of the 18 languages supported by RPAS.

These sections describe the integration configuration:

- Shared Hierarchies
- Shared Facts
- Integration Map
- Domain and Configuration Map
- Integration Configuration Format

Shared Hierarchies

The shared hierarchy section defines the hierarchical structure of the RDM and it follows similar xml structure to the hierarchy.xml in a domain but has been simplified.

This section consists of multiple hierarchies, each forming a dimension graph. Each hierarchy has a name, a label and an order number which is used to define the order of multiple dimensions in an intersection. Each dimension has a name and a label, and the parent-child relationship of the xml structure defines the aggregation and spread relationship of the dimensions.

Shared Facts

A fact in RDM is the corresponding data entity of a measure in a domain. The properties for a fact are name, label, base intersection, type, fact table, NA value, description, and purge age.

If a fact is shared by the RDM, a fact group name must be specified. Otherwise, the fact is shared by domains only.

Integration Map

Each entry defines the mapping between one domain measure and one database fact.

Domain and Configuration Map

This section defines the domain and configuration map. Each domain is mapped to a domain configuration. The domain name is configured here which is by default the same as the configuration name. This section is used internally by the Integration tool of the ConfigTools.

Integration Configuration Format

Example 8–2 shows the format of the integration.xml file which contains all of the xml elements described in the previous sections.

Example 8–2 Format for integration.xml

```
RDM.xml
<?xml version="1.0" encoding="UTF-8" ?>
<rpas_hsa_configuration name="RDM" language=”English”>
<shared_hierarchy_set>
   <hierarchy name="CLND">
      <label>Calendar</label>
      <order>9990</order>
      <dimension name="day">
         <label>Day</label>
         <position_format>d%Y%m%d</position_format>
         <dimension name="week">
     <label>Week</label>
        <dimension name="mnth">
    <label>Fiscal Month</label>
       <dimension name="qrtr">
<label>Fiscal Quarter</label>
   <dimension name="year">
<label>Fiscal Year</label>
   </dimension>
</dimension>
     </dimension>
         </dimension>
      </dimension>
   </hierarchy>
   <hierarchy name="PROD">
      <label>Product</label>
      <order>9980</order>
      <dimension name="sku">
         <label>Sku</label>
         <dimension name="clss">
            <label>Class</label>
               <dimension name="dept">
```

```
           <label>Department</label>
            </dimension>
            </dimension>
  </dimension>
   </hierarchy>
</shared_hierarchy_set>
<shared_fact_set>
   <fact name="P">
      <label>P Fact</label>
      <baseint>sku_week</baseint>
      <type>2</type>
      <fact_group>r_g_sku_week_ft</fact_group>
      <navalue>0</navalue>
      <description>PPP Fact</description>
      <purgeage>1000</purgeage>
   </fact>
   <fact name="Q">
      <label>Q Fact</label>
      <baseint>sku_week</baseint>
      <type>4</type>
      <fact_group>r_g_sku_week_ft</fact_group>
      <navalue>0</navalue>
      <description>QQQ Fact</description>
      <purgeage>1000</purgeage>
   </fact>
</shared_fact_set>
<integration_map>
   <entry>
      <domain>d1</domain>
      <measure>a</measure>
      <fact>P</fact>
   </entry>
   <entry>
      <domain>d2</domain>
      <measure>a</measure>
      <fact>P</fact>
   </entry>
   <entry>
      <domain>d1</domain>
      <measure>c</measure>
      <fact>Q</fact>
   </entry>
</integration_map>
<domain_set>
   <domain name="d1">
      <config_path>...</config_path>
   </domain>
   <domain name="d2">
      <config_path>...</config_path>
   </domain>
   <domain name="d3">
      <config_path>...</config_path>
   </domain>
</domain_set>
</rpas_hsa_configuration>
```

Partition Information

The RDM can be built to make use of database partitioning. This partitioning enables parallelization within the RDM in a manner similar to a RPAS domain. As with domain partitioning, information about the desired partitioning scheme is contained with a partition information document. The RDM partition information xml file is modeled after the global domain configuration. It contains the partition dimension name and multiple partition definitions, each with a partition name and its corresponding list of positions.

This xml file can be generated by the `rdmPartitionMgr` utility. for more information on the `rdmPartitionMgr` utility, refer to either the *Oracle Retail Predictive Application Server Administration Guide for the Fusion Client* or the *Oracle Retail Predictive Application Server Administration Guide for the Classic Client*.

RDM SSL Configuration

In an RPAS HSA environment, RPAS processes access shared data stored in the RDM through a network connection. Oracle database has built-in support for secure connection through Secure Socket Layer (SSL). RPAS HSA is taking advantage of this Oracle feature and provides support for network security through SSL. This section covers the configuration of SSL for RDM, including setting up SSL for both the Oracle database server and the Oracle client used by RPAS processes. Note that the SSL configuration of RDM connection is optional in HSA.

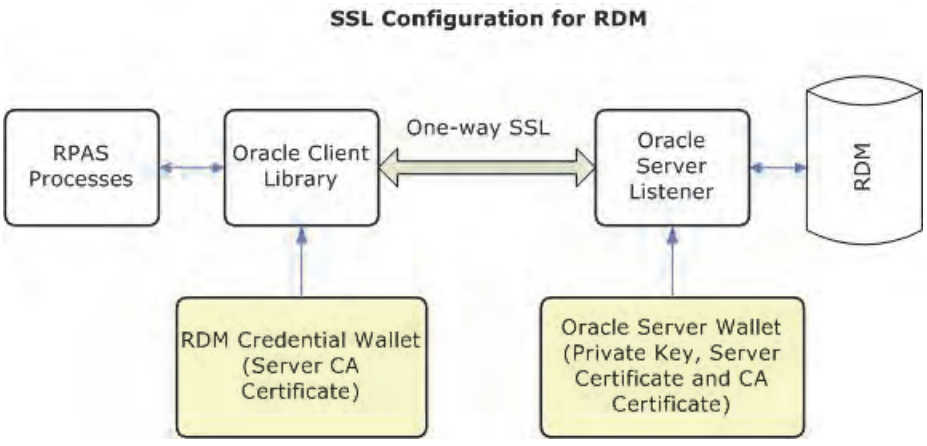
Figure 8–7 shows the components of the SSL configuration for RDM. The client/server connection is protected by one-way SSL, which only authenticates the server on the client side. As a result, only the Oracle Server Wallet is required to have a private key. The client wallet will only need to contain the server CA (Certificate Authority) certificate.

The CA certificate can be either self-signed or issued by a third-party CA. In the case of third-party CA, more than one certificate may need to be imported into the client wallet (usually called a CA certificate chain).

The server side configuration is mostly done manually by the DBA. A script is provided to create the wallet if a self-signed certificate is used.

The client side configuration is done through scripts by the RPAS admin.

Figure 8–7 SSL Configuration for RDM



Setting Up SSL on the Oracle Server

These sections contain information on setting up SSL on the Oracle server side:

- Creating the Oracle Server Wallet
- Updating the Oracle Server Network Configuration
- Updating the Oracle Listener Configuration

> **Note:** Setting up SSL should be done by a DBA who has the permission to modify the configuration files of the Oracle server.

Creating the Oracle Server Wallet

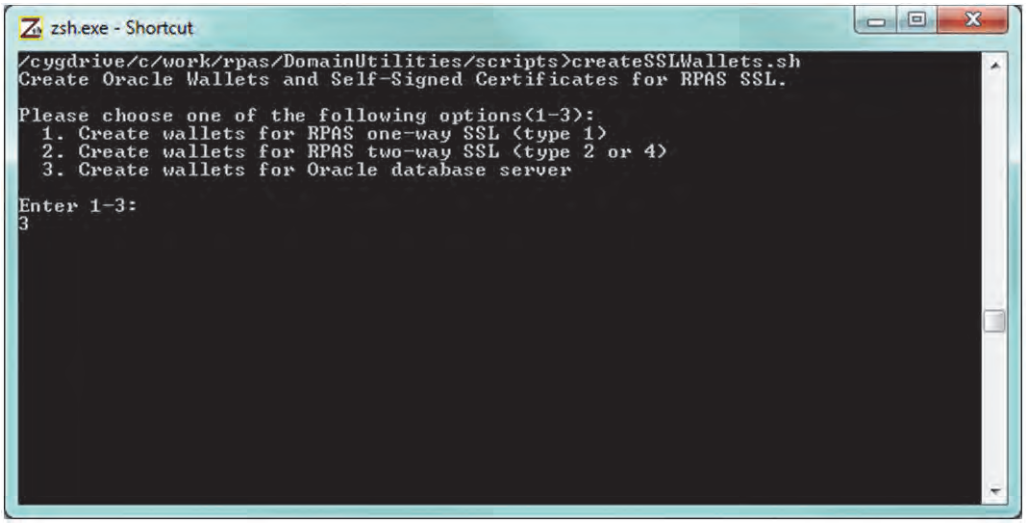
The Oracle server wallet can be created the same way as is the RPAS server wallet for one-way SSL that is described in the SSL chapter of either the *Oracle Retail Predictive Application Server Administration Guide for the Fusion Client* or the *Oracle Retail Predictive Application Server Administration Guide for the Classic Client*.

On UNIX, the wallet must be created by the user account which starts the Oracle database processes. For security reason the wallet is only accessible by its creator.

If a self-signed CA certificate is used, then the RPAS provided shell script, `createSSLWallets.sh` creates the root and server wallets, private keys, and all related certificates.

When running the script `createSSLWallets.sh`, choose Option 3 to create wallets for Oracle database server, as shown in Figure 8–8.

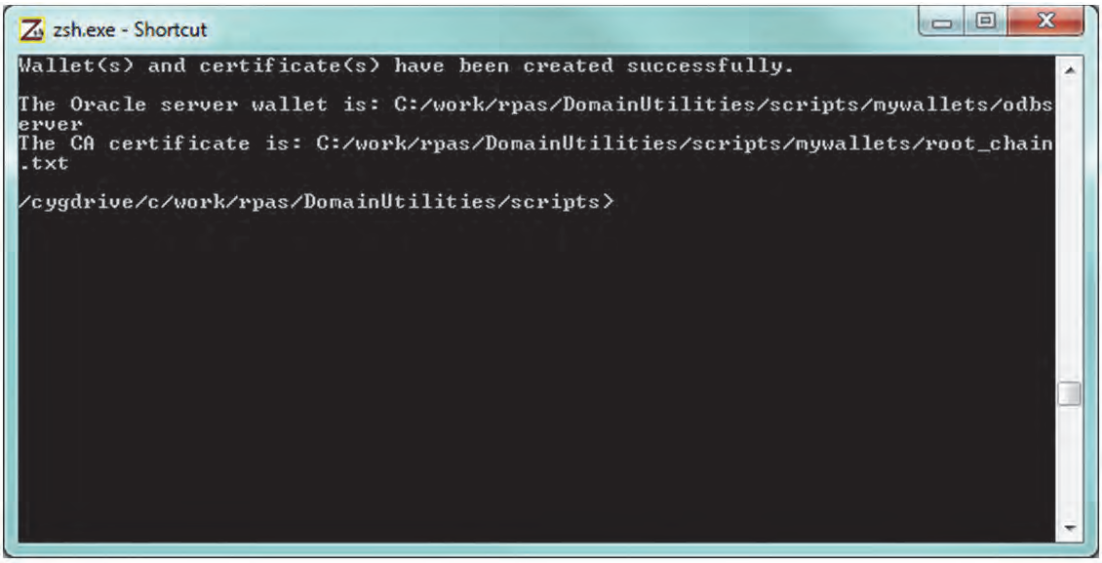
Figure 8–8 Creating Oracle Wallets for RPAS SSL



The script asks for the root directory where the wallets are to be created, your organization name, and passwords for the root and server wallets. At the end of a successful run, the wallet location and the path to the CA certificate file are displayed as shown in Figure 8–9.

The CA certificate file can be copied to a common location and then needs to be imported into the Oracle client wallet.

Figure 8–9 Wallet Location and the Path to the CA Certificate File



Updating the Oracle Server Network Configuration

The network configuration file `sqlnet.ora` normally resides in the **$ORACLE_HOME/network/admin** directory.

The following table lists the required settings for SSL

> **Note:** The SSL client authentication parameter must be set to FALSE for one-way SSL. The `wallet_location` is the absolute path to the Oracle server wallet as described in Creating the Oracle Server Wallet.

Required SSL Settings in sqlnet.ora on the Oracle Server

```
SSL_CLIENT_AUTHENTICATION = FALSE
WALLET_LOCATION =
 (SOURCE =
  (METHOD = FILE)
  (METHOD_DATA =
   (DIRECTORY = wallet_location)
  )
 )
```

The following table lists the optional settings for SSL.By default, Oracle secure connection uses SSL version TLS 1.0 which can be overridden by the SSL_VERSION setting.

> **Note:** For more information regarding these settings, refer to the Oracle Database Security Guide at:
>
> `http://docs.oracle.com/database/121/DBSEG/toc.htm`

Optional SSL Settings in sqlnet.ora on the Oracle server
SSL_CIPHER_SUITES= (SSL_cipher_suite1 [,SSL_cipher_suite2])
SSL_VERSION=1.2

Updating the Oracle Listener Configuration

The listener configuration file `listener.ora` normally resides in **$ORACLE_HOME/network/admin** directory. After modification, the Oracle listener must be restarted for the changes to take effect.

Note: The wallet location and SSL settings are the same as that in the network configuration file (must be in both files). A new listener endpoint with protocol TCPS must be added to the LISTENER setting. Port number 2484 is the typical port for TCPS but other number can also be used, while the existing protocol listings of the LISTERNER setting can be kept as they are.

Required SSL Settings in listener.ora on the Oracle server

```
SSL_CLIENT_AUTHENTICATION = FALSE
WALLET_LOCATION =
 (SOURCE =
  (METHOD = FILE)
  (METHOD_DATA =
   (DIRECTORY = wallet_location)
  )
 )
LISTENER =
 (DESCRIPTION_LIST =
  (DESCRIPTION =
   (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1521))
  )
  (DESCRIPTION =
   (ADDRESS = (PROTOCOL = TCP)(HOST = odbserver-host)(PORT = 1521))
  )
  (DESCRIPTION =
   (ADDRESS = (PROTOCOL = TCPS)(HOST = odbserver-host)(PORT = 2484))
  )
 )
```

Setting Up SSL on the Oracle Client

These sections contain information on setting up SSL on the Oracle client side for use by RPAS or RDM processes:

- Updating Schema Information Configuration
- Importing the Sever CA Certificate
- Updating Oracle Client Network Configuration
- Updating Oracle Net Service Names

Note: Only Updating Schema Information Configuration is performed by the user. The other three steps are processed by the `prepareRDM.sh` script.

Updating Schema Information Configuration

The schema information configuration must have matching endpoint parameters to the Oracle listener. The protocol must be `tcps` and the port number must be the same as specified in the listener configuration.

The following table lists the TNS parameters in `schemaInfo.xml`.

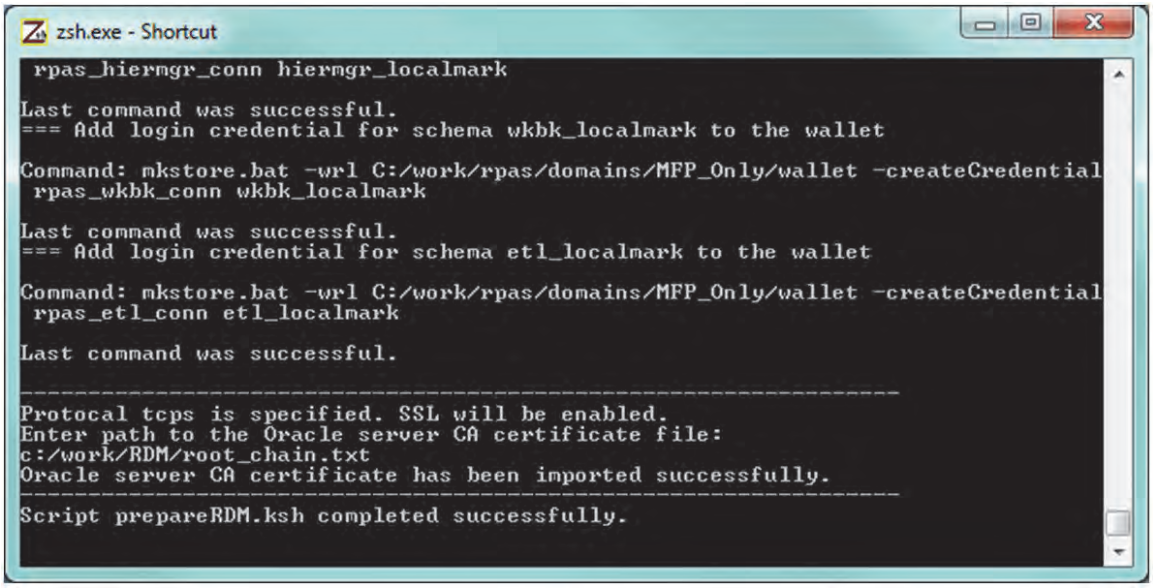
TNS parameters in schemaInfo.xml
<tns_parameters> <protocol>tcps</protocol><host>odbserver-host</host><port>2484</port><server>dedicated</server><service_name>myservice</service_name><sid></sid>

Importing the Sever CA Certificate

The `prepareRDM.ksh` script in the RDM build process already creates an Oracle wallet to hold the login credentials for RDM. The same wallet should be used to hold the Oracle server CA certificate. If protocol `tcps` is specified in the schema information, this same script asks for the path to the CA certificate file and imports it into the wallet automatically.

Figure 8–10 shows the importing of CA certificate by the `prepareRDM.ksh` script.

Figure 8–10 Importing the Sever CA Certificate



If a third-party CA is used and there is more than one file in the CA certificate chain, then you must specify the top certificate for the script to import and then manually import the rest in the order of the chain using the following command:

```
orapki wallet add -wallet {client_wallet_directory} -trusted_cert -cert
{ca_cert_chain_file} -pwd {client_wallet_password}
```

Note: The client wallet location is the **wallet** subdirectory of the RDM repository.

Example 8–3 Manual Importing the Sever CA Certificate Command

```
orapki wallet add -wallet C:/wallets/client -trusted_cert -cert C:/wallets/ca_
chain2.txt -pwd clientpass1
```

Updating Oracle Client Network Configuration

The network configuration file `sqlnet.ora` used by RPAS processes is created automatically by the RDM Manager during the RDM build process. This file resides in the **tns_admin** subdirectory of the RDM repository.

The following table lists the required SSL settings in `sqlnet.ora` on the Oracle client.

> **Note:** The SSL client authentication parameter must be set to TRUE for one way SSL. The `wallet_location` is the **wallet** subdirectory of the RDM repository.

Required SSL Settings in sqlnet.ora on the Oracle client

```
SSL_CLIENT_AUTHENTICATION = TRUE
WALLET_LOCATION =
 (SOURCE =
  (METHOD = FILE)
  (METHOD_DATA =
   (DIRECTORY = wallet_location)
  )
 )
```

Updating Oracle Net Service Names

The Oracle Net Service Names configuration file `tnsnames.ora` is created automatically by the RDM Manager during the RDM build process in the **tns_admin** subdirectory of the RDM repository. All entries use the endpoint parameters specified by the schema information configuration.

Net Service Names using SSL in tnsnames.ora on the Oracle client

```
rpas_data_mart_conn =
 (DESCRIPTION =
  (ADDRESS_LIST =
   (ADDRESS = (PROTOCOL = tcps)(HOST = odbserver-host)(PORT = 2484))
  )
  (CONNECT_DATA =
   (SERVER = dedicated)
   (SERVICE_NAME = myservice)
  )
 )
.
.
.
```

Test and Confirm the SSL Connection

After SSL configuration is complete on both the Oracle server and client and the Oracle listener has been restarted, then use RDM Manager to verify the connection using this command:

```
rdmMgr -rdm rdmPath -testConnection
```

Note: There is a small delay, sometimes a few minutes, after the listener is restarted and the SSL connection is up

RPAS Installation Utilities

These sections describe the main RPAS installer and all helper utilities/scripts:

- RDM Preparation Script
- RPAS Installer
- RDM Manager

RDM Preparation Script

A shell script is created to configure the RDM before building the database schemas. The script is interactive but can be made silent if all parameters are specified by environment variables.

Note: If SSL is used for the RDM database connection, then it must first be configured prior to running this script.

Script

```
prepareRDM.ksh    [ rdmPath]    [schemaInfoFile]
```

Functions

This script performs the following functions:

1. Creates the RDM file system repository.
   - Calls `rdmMgr` to create the RDM repository
2. Generates RDM pre-build scripts and network configuration files.
   - Calls `rdmMgr -genPreBuildScript` to create the preinstallation script and network configuration files.
   - The output files are placed in the RDM repository:

     The file, `prepareSchemas.sql` is placed in the RDM repository subdirectory **scripts**. Passwords are prompted for while running this SQL file.

     The files, `tnsnames.ora` and `sqlnet.ora` are placed in the RDM repository subdirectory **tns_admin**.
3. Creates an Oracle wallet for RDM.
   - Creates an Oracle wallet named `rdmPath/wallet`
4. Adds all RDM user credentials in the wallet.
   - Passwords are prompted for these schema logins.

Note: The password for each schema in the wallet must match to that of the corresponding schema created in the database.

RPAS Installer

The existing `rpasInstall` program is enhanced to support building an HSA system.

Command Line Syntax

The following example displays the command line syntax for `rpasInstall`.

Example 8–4 Command Line Syntax for rpasInstall

```
rpasInstall {command} {options}
rpasInstall
rpasInstall -buildRDM -genScript -integrationCfg <integration_config>
{-partitionInfo <partitionInfo_file> | -noPartition} -log <log_file> -rdm <rdm_
path>
rpasInstall -buildRDM -integrationCfg <integration_config> {-partitionInfo
<partitionInfo_file> | -noPartition} -rdm <rdm_path> -log <log_file>
rpasInstall -buildRDM -fromScript -rdm <rdm_path> -log <log_file>
rpasInstall -rebuildRDM -integrationCfg <integration_config> {-partitionInfo
<partitionInfo_file> | -noPartition} -rdm <rdm_path> -in <input_directory> -log
<log_file>
rpasInstall -rebuildRDM -fromScript -rdm <rdm_path> -log <log_file>
rpasInstall {command} {options}
```

Table 8–4 describes the command and option values.

Table 8–4 Command and Option Values for rpasInstall

Value	Description	Type
-buildRDM	Performs a build installation for integration deployment.	Command
-rebuildRDM	Performs a rebuilt installation for integration deployment.	Command
-integrationCfg <config_file>	Option for the integration configure file.	Option
-partitionInfo <partitionInfo_file>	Option for the partition information input file.	Option
-noPartition	Option if no partition information is provided.	Option
-rdm <path>	Option for the RDM repository path.	Option
-genScript	Only generates the SQL script with RDM schema definitions, does not deploy it. Two script files named `buildDataMart.sql` and `buildUserSchemas.sql` are generated in the subdirectory **scripts** of the RDM repository. One file script is to create Data Mart schema's schema objects and other is for other users' schema objects.	Option
-fromScript	Builds or patches RDM using the scripts `buildDataMart.sql` and `buildUserSchemas.sql` in the RDM repository subdirectory **scripts**. This option enables the database administrator to customize the RDM creation statements and build.	Option
-log <log_file>	For logging - all logging output from the sub-processes should be re-directed to this file. At the end of the `rpasInstall` run, a copy of this file should be copied to `{rdm_repository}/output/rpasInstall` with *timestamp* in the file name.	Option
-verbose	Results in more detailed log information.	Option

RDM Manager

The RDM Manager is a command line utility which manages operations on the RDM. It creates the RDM repository directory structure and the RDM database schema from integration configuration and optional partition information. It also manages domain registration and provides some other useful tools to support the RDM in an integrated environment.

These sections describe setting up the RDM Manager:

- Registering a Domain with RDM
- Unregistering a Domain from RDM
- Configuration Files and Output Files
- Command Line Syntax
- Creating the RDM Repository
- Building RDM Schema
- Testing the ODB Connection
- Registering a Domain with RDM
- Unregistering a Domain from RDM
- Listing Properties or Domains

Registering a Domain with RDM

The command, `rdmMgr`, registers a domain at a time. Prerequisites to registering a domain include:

- The partition level of the domain must be at or upper to the RDM partition level.
- The domain to be registered with the RDM must have compatible hierarchies.
- The domain's all informal positions must be formalized or removed.

Registering a domain does the following:

- Validates the domain hierarchy's compatibility with the RDM.
- Adds and assigns a new RDM property, `domain_name`, in the domain.
- Registers the domain name with RDM in a domain table. An integer domain ID is created, which is used internally in the RDM.
- Updates the ITT tables in the RDM with the domain's positions.

Unregistering a Domain from RDM

The command, `rdmMgr`, unregisters a domain at a time.

> **Note:** Before unregistering the domain, it is recommended to transfer shared measure data belonging to the domain. It can be transferred to the domain using the transferFactdata utility.

Unregistering a domain does the following:

- Removes the domain name from the domain table in the RDM.
- Removes all RDM properties created during registration from the departing domain.
- Removes all entries in the ITT tables which belong to the departing domain.
- Formalizes the informal positions of the departing domain to keep the measure data.
- Preserves the departing domain data.

Configuration Files and Output Files

The integration configuration file and partition information file are not specified on the command lines. They should be pre-staged in the **config** directory of the RDM repository, usually by the `rpasInstall` process. The scripts and network configuration files generated by the RDM manager all have pre-defined output directories and names.

Pre-staged Configuration Files

- Integration configuration

  Copy to `rdmPath/config/integrationCfg.xml` in advance.

- Partition information

  Copy to `rdmPath/config/partitionInfo.xml` in advance.

Script Output Files

All output scripts to go to this directory: **rdmPath/scripts/**

These scripts have fixed names:

Name	Type	Description
prepareSchemas.sql	Pre-build script	SQL script to create users and corresponding roles with few basic privileges.
buildDataMart.sql	Build script	SQL script to create private synonyms.
buildUserSchemas.sql	Build script	This script should be run for each access schema other than the data mart schema.
grantPrivileges.sql	Build script	SQL script to grant all required privileges to each access schemas. This should be re-executed only when the schema objects are created for the first time or when existing schema objects are re-created.

Oracle Database Network Configuration Files

The following files go to this directory: **rdmPath/tns_admin**:

- tnsnames.ora
- sqlnet.ora

> **Note:** The environment variable TNS_ADMIN should be assigned the path to the **tns_admin** directory of the RDM repository, which is being done programmatically in the C++ code.

Command Line Syntax

The following example displays the command line syntax for the RDM Manager.

Example 8–5 Command Line Syntax for the RDM Manager

```
rdmMgr -rdm {rdmPath} {command} [options]
```

Commands called by:	Command Line Syntax
prepareRDM.ksh script	rdmMgr -rdm rdmPath -create -schemaInfo f1
	rdmMgr -rdm rdmPath -genPreBuildScript
RPAS Installer	rdmMgr -rdm rdmPath -clearSchema
	rdmMgr -rdm rdmPath -buildSchema
	rdmMgr -rdm rdmPath -genBuildScript
	-buildFromScript
Commands used by the:	**Command Line Syntax**

Commands called by:	Command Line Syntax
RPAS Administrator	rdmMgr -rdm rdmPath -listProperties
	rdmMgr -rdm rdmPath -testConnection
	rdmMgr -rdm rdmPath -register -d domainPath -name domainName
	rdmMgr -rdm rdmPath -unregister -name domainName
	rdmMgr -rdm rdmPath -listDomains
	rdmMgr -rdm rdmPath -listHistory
	rdmMgr -rdm rdmPath -purgeDeletedData
	rdmMgr -rdm rdmPath -updateRegistration -d domainPath -name domainName
	rdmMgr -rdm rdmPath -move -dest destRdmPath

Table 8–5 describes the command and parameter values.

Table 8–5 Command and Parameter Values for the RDM Manager

Value	Description	Type
-create	Creates the RDM repository directory structure and copies the `schemaInfo` file to the **config** directory of the RDM repository.	Command
-listProperties	Lists the RDM properties.	Command
-testConnection	Tests all schema connections.	Command
-clearSchema	Drops any existing RDM schema objects in all the schemas (without access).	Command
-buildSchema	Builds the RDM schema objects in all schemas (without access) and grant privileges to the roles created initially. Assuming `integrationCfg.xml` and `partitionInfo.xml` are present in the **config** directory of the RDM repository.	Command
-genPreBuildScript	Generates the SQL script that can be used to create schema user and roles. The generated script `prepareSchemas.sql` is in the **scripts** directory of the RDM repository.	Command
-genBuildScript	Generates the SQL script that creates the schema objects, populates some metadata, and grants privileges to the roles created initially. Assuming `integrationCfg.xml` and `partitionInfo.xml` are present in the **config** directory of the RDM repository. The generated scripts `buildDataMart.sql`, `buildUserSchemas.sql` and `grantPrivileges.sql` are in the **scripts** directory of the RDM repository	Command
-buildFromScript	Builds the RDM schema objects and grants privileges using the SQL scripts `buildDataMart.sql`,`buildUserSchemas.sql` and `grantPrivileges.sql` generated by `genBuildScript`.	Command
-register	Registers the RPAS domain with RDM.	Command
-unregister	Unregisters the RPAS domain from RDM.	Command
-listDomains	Lists the current domains and status.	Command
-listHistory	Lists the RDM schema and RDM Repository history	Command
-purgeDeletedData	Physically purges all dimension data that are marked for deletion, along with all associated fact data.	Command
-updateRegistration	Updates existing registration of a domain. Depending on what has changed or become out of sync, it updates the domain path, RDM path, ITT tables, or all.	Command

Table 8–5 (Cont.) Command and Parameter Values for the RDM Manager

Value	Description	Type
-move	Moves the RDM repository to a new location and updates all domains with the new RDM path. If any of the domains cannot be updated, a warning message displays. To update the RDM path in the domain separately, run the `-updateRegistration` command.	Command
-d	Path to the domain.	Parameter
-rdm	Path to the RDM.	Parameter
-schemaInfo	Path to the schema information file This file is copied to the **config** directory of the RDM repository.	Parameter
-name	The domain name as defined in the integration configuration.	Parameter
-dest	The destination path for moving RDM repository.	Parameter

Creating the RDM Repository

This process results in the:

- Creation of the RDM Repository directory structure
- Saving the schema names in `schemaInfo.xml` as RDM properties

Building RDM Schema

This process creates these RDM objects:

- Dimension tables
- Fact tables
- Metadata Tables
- Data flow table
- Index Translation Tables (ITT)
- Commit, logging and dimension staging tables
- Other ODB objects (sequences, views, indexes, types, PL/SQL packages, and so on)

> **Note:** Additionally, private synonyms for each schema user are also created.

Testing the ODB Connection

The connections to all ODB schemas are tested using this command: `rdmMgr -testConnection`

Registering a Domain with RDM

The command, `rdmMgr`, registers a domain at a time. Prerequisites to registering a domain include:

- The partition level of the domain must be at or upper to the RDM partition level.
- The domain to be registered with the RDM must have compatible hierarchies.
- The domain's all informal positions must be formalized or removed.

Registering a domain does the following:

- Validates the domain hierarchy's compatibility with the RDM.
- Adds and assigns a new RDM property, `domain_name`, in the domain.
- Registers the domain name with RDM in a domain table. An integer domain ID is created, which is used internally in the RDM.
- Updates the ITT tables in the RDM with the domain's positions.

Unregistering a Domain from RDM

The command, `rdmMgr`, un-registers a domain at a time.

> **Note:** Before unregistering the domain, it is recommended to transfer shared measure data belonging to the domain. It can be transferred to the domain using the `transferFactdata` utility.

Unregistering a domain does the following:

- Removes the domain name from the domain table in the RDM.
- Removes all RDM properties created during registration from the departing domain.
- Removes all entries in the ITT tables which belong to the departing domain.
- Formalizes the informal positions of the departing domain to keep the measure data.
- Preserves the departing domain data.

Listing Properties or Domains

Figure 8–11 shows the output of `rdmMgr -listDomains`. Two blank lines indicate the end of output.

Figure 8–11 Output of rdmMgr -listDomains

```
Domain Name     Path

-----------------------------------------------

RPAS_UT         /vol.nas/rpas_se/RPAS_UT


<two blank lines>

rdmMgr ran successfully
```

Figure 8–12 shows the output of `rdmMgr -listProperties`. Two blank lines indicate the end of output.

Figure 8–12 Output of rdmMgr -listProperties

```
Property Name      Value

-------------------------------------------

rpas_data_mart:           rpas1

rpas_wkbk_user:            whatever

...

<two blank lines>

rdmMgr ran successfully
```

Part II

Patch Installation

Part II of this guide details the steps needed to perform a patch installation of RPAS.

Part II contains the following chapters:

Chapter 9, "RPAS Package Extraction"

Chapter 10, "RPAS Server Patch Installation"

Chapter 11, "RPAS Fusion Client Patch Installation"

Chapter 12, "RPAS Classic Client Patch Installation"

Chapter 13, "HSA Patch Installation"

For information about a full installation, refer to Full Installation

Upgrading Process

Releases earlier than RPAS 13.3 have special considerations that need to be followed when upgrading. Previous upgrade instructions do not apply. It is very important that you follow the process as detailed in the following table.

The RPAS Upgrade process is dependent upon your current RPAS version.

Current Version	Upgrade Process
Pre-12.1	At the time of the 13.3.1 release, Oracle Retail is aware of customer implementations that are still lower than version 12.1. Oracle Retail advises such customers to be sure to follow the process outlined in the 12.1 version of the *Oracle Retail Predictive Application Server Installation Guide* to convert their pre-12.1 domain to a post-12.1 domain. Fundamental changes have been made to RPAS's storage layer, and pre-12.1 domains are not upgradable to 13.3 domains simply with the use of the upgradeDomain utility. After following the upgrade instructions to convert the environment to 12.1, customers can then continue with an upgrade to 13.3.1, which also has special instructions as listed in the following row.
Pre-13.3	RPAS 13.3.1 requires a special upgrade process even if you are on major version 13. After installing the code described in "Part II: Patch Installation" of this guide, proceed to either the Classic Client or Fusion Client version of the *Oracle Retail Predictive Application Server Administration Guide*. The chapter, "Building and Upgrading Domains ", details the full process to convert and upgrade your domain environment to be compatible with RPAS 13.3.1.
13.3 and later	If you are already using RPAS 13.3, there is no special upgrade process to follow. Continue the regular patch installation as described in the "Part II: Patch Installation chapters" of this guide.

9

RPAS Package Extraction

The first step in upgrading to the most recent installation is to download the 14.1.2 release from the My Oracle Support Web site (https://support.oracle.com) to a staging folder (such as $PACKAGEDIR) that is accessible to all components of your current RPAS environment.

Example Package Extraction

The following example describes a sample upgrade installation. These sample commands are provided to guide you through the file extraction process and to identify the files provided in this upgrade.

```
$ mkdir packagedir
$ cp rpas.zip packagedir
$ cd packagedir
$ export PACKAGEDIR=`pwd`
$ unzip rpas.zip
```

The following items may be extracted to the current directory:

```
ARPOPlatform-14.1.2.aix71.tar.zip
ARPOPlatform-14.1.2.aix71.tar.zip
ARPOPlatform-14.1.2.sun10.tar.zip
ARPOPlatform-14.1.2.linux.tar.zip
ARPOPlatform-14.1.2.nt.zip
ARPOPlatform-14.1.2.clients.zip
Curve14.1.2.zip
Grade14.1.2.zip
FusionClient.zip
README.html
```
DOCS folder

Note: The file, ARPOPlatform-14.1.2.clients.zip, is an archive of the RPAS Classic Client and ODBC Client for all platforms.

The Curve and Grade documentation sets which were previously included in the **DOCS** folder are now combined with the Oracle Retail Demand Forecasting (RDF) documentation. This documentation is available within the RDF package and on the following Web site:

http://www.oracle.com/technology/documentation/oracle_retail.html

---

At this point, you must choose which package you wish to extract, based on your current server platform and version. AIX 7.1 is used for the purpose of the following example.

Next, run the following commands.

```
$ unzip ARPOPlatform-14.1.2.aix71.tar.zip
```

```
$ tar -xf ARPOPlatform-14.1.2.aix71.tar
```

Now the package directory should contain a subdirectory named **ARPOPlatform**. You have successfully completed extracting the upgrade.

10

RPAS Server Patch Installation

This chapter describes how to patch the RPAS Server on UNIX or Windows. For instructions on patching the server on Windows, refer to Appendix C, "Appendix: rsp_manager Usage".

RPAS Upgrade Prerequisites

In order to upgrade RPAS, first verify the following criteria for the RPAS system:

- Verify that RPAS is currently installed.
- Verify that UNIX operating system is updated to the currently supported version, which can be found in the section, Hardware and Software Requirements.
- Verify that the environment variables are correctly set; if they are not, follow these instructions to set them:
  - On Windows, update the environment variables using the instructions in the section, Creating the Required Environment Variables.
  - On UNIX, change directories to the original RPAS installation directory (such as the one created by the most recent installer), and run `retaillogin.ksh` to set all environment variables. For example:

    ```
    $ cd /retail
    $ . ./retaillogin.ksh
    ```

    Note: Once you have run the script, verify that the environment variables all point to the correct locations on your environment.

    If you have updated Java since the last installation of RPAS, verify that the JAVA_HOME path is correct. If not, update your `RIDE_HOME/toolslogin.ksh` script and source your `retaillogin.ksh` again as previously outlined.

Upgrade to Key RPAS Versions

Before you upgrade to a post-13.3 release, ensure that you have upgraded and patched to RPAS Release 13.2.3.

Upgrade Process to RPAS Release 13.3

This list defines the upgrade process that is described in the following sections.

1. Upgrade and Patch to RPAS Release 13.2.3

2. Convert for Integer Indexing
3. Upgrade and Patch to RPAS Release 13.3 or Later

Upgrade and Patch to RPAS Release 13.2.3

Upgrading your domain to a 13.3 or later RPAS domain requires that you first upgrade it to a 13.2.3 domain. For instructions, see the "Upgrading and Patching Domains" section in the 13.2.3 release of either the Classic Client or Fusion Client version of the *Oracle Retail Predictive Application Server Administration Guide*.

After upgrading, you must run a configuration patch over the domain to ensure compatibility with the RPAS version, regardless of whether there are any configuration changes.

Convert for Integer Indexing

Upgrading to a 13.3 or later RPAS domain requires the use of the convertDomain utility, which may not copy the entire contents of the source domain to the destination domain. For files that are not copied by convertDomain, you may need to copy them manually.

For information about what is and is not copied, see the "convertDomain" section in the 13.3 release of either the Classic Client or Fusion Client version of the *Oracle Retail Predictive Application Server Administration Guide*.

Upgrade and Patch to RPAS Release 13.3 or Later

For instructions, see the "Upgrading and Patching Domains" section in the 14.1 release of either the Classic Client or Fusion Client version of the *Oracle Retail Predictive Application Server Administration Guide*.

After upgrading, you must run a configuration patch over the domain to ensure compatibility with the RPAS version, regardless of whether there are any configuration changes.

For Windows: Install Cygwin

Cygwin is required if the RPAS Server is to be installed on Windows operating systems. You can find more information about downloading this product at `http://www.cygwin.com.`

For installation guidance, refer to the following note on My Oracle Support (`https://support.oracle.com`):

Cygwin Setup Guide for RPAS (Note 1333398.1)

This document details necessary options you should select when installing Cygwin to make it function properly with RPAS.

Java Environment

Ensure that Java Development Kit (JDK) has been installed on the machine where RPAS will run and that the JAVA_HOME environment variable is properly set.

For detailed Java environment information, refer to the section, Java Environment, in the Installing on UNIX and Linux Environments chapter.

Ride Options

The RIDE_OPTIONS environmental variable has been defined to allow users to pass information into the rpasInstall process. Unlike the regular arguments passed on the command line to rpasInstall (such as -fullinstall and -updatestyles), arguments defined in RIDE_OPTIONS are passed to every rpasInstall instance that runs in the environment.

There the three supported properties for use with RIDE_OPTIONS:

Property	Used for
Xmx	Java
HP 64-bit mode Java (-d64)	HP Itanium and Solaris
Drpas.maxProcesses	RPAS

For Java

Xmx

By default, the Java Virtual Machine requests on the order of 268 MB of RAM from the OS to allocate for its heap. Large domains that are built from complex configurations can potentially exhaust this limited amount of memory. This is even more of an issue in patch installations than in builds since a patch installation requires two different versions of a configuration to be held in memory simultaneously.

By using the `-Xmx` option, you can instruct the Java Virtual Machine to request more memory from the OS to prevent situations when all allocated memory is exhausted. The syntax of the property is:

`-Xmx###m`, where ### is the amount, in megabytes, of memory the JVM is to request. Common values for this argument are `-Xmx512m` or `-Xmx1024m`.

For HP Itanium and Solaris

HP 64-bit mode Java (-d64)

The HP distribution of Java does not consist of separate executables for 32-bit and 64-bit operating systems. Instead, there is a single distribution of Java that can run in either 32-bit or 64-bit mode. By default, the HP Java runs in 32-bit mode. Because RPAS is built as a 64-bit executable on the Itanium OS, the RPAS libraries are unable to link with Java if it is running in 32-bit mode.

By adding the `-d64` property to RIDE_OPTIONS, the HP Java distribution is 64-bit mode enabled and the RPAS libraries link successfully.

It is often the case that users may want to use or more different properties in conjunction with RIDE_OPTIONS. When this is the case, all desired properties should be included within the environmental variable definition separated by white space with the entire definition enclosed in double quotes.

An example of this is shown as:

```
export RIDE_OPTIONS="-d64 -Xmx1024m -Drpas.maxProcesses=8"
```

For RPAS

Drpas.maxProcesses

Several RPAS server utilities are designed to take advantage of multi-processor hardware to improve their performance. These utilities attempt to perform operations in parallel, each process running on a distinct processor. The

The `-Drpas.maxProcesses` argument is used to instruct RPAS how many processors it should attempt to run in parallel when running one of the server utilities that has multi-processor support when that utility is run as a part of the rpasInstall process.

Note that the `-Drpas.maxProcesses` argument only affects those calls to server utilities made from within the rpasInstall process and does not affect calls to server utilities made from the command line or as part of a batch job.

The syntax of the property is:

```
-Drpas.maxProcesses=###
```

Where ### is the number of sub-processes the RPAS server utility should attempt to run in parallel. The number of processes to use should be determined by the administrator of the hardware system based on the physical number of processors available and the amount of load that is acceptable for the rpasInstall process to place on the system.

RPAS Upgrade Process

The following process outlines how to upgrade the RPAS server environment to the current version.

Note: The version number for RPAS may include a fifth decimal position that refers to a build number.

---

1. In a command prompt, change to location of the base directory of this upgrade.

   ```
   $ cd $PACKAGEDIR/ARPOPlatform/14.1.2
   ```

2. Run RSP Manager to upgrade your environment:

   If the platform is Linux or Cygwin on Windows, use the following command:

   ```
   $ ./rsp_manager.linux –install –sp linux –no_domain
   ```

   For all other platforms, use the following command:

   ```
   $ ./rsp_manager –install –sp [PLATFORM] –no_domain
   ```

   Note: [PLATFORM] represents your current platform and should be replaced with the correct label, such as aix71.

   `-no_domain` indicates that there no domain in need of upgrading.

   If you want to upgrade a single domain, include `–domain` with the full name or path of the domain. If you want to upgrade multiple domains at the same time, you must include `–domain` and pass on a file that contains a list of all domains that need to be upgraded.

For instructions on upgrading domains, refer to the "Building and Upgrading Domains" chapter in either the Classic Client or Fusion Client version of the *Oracle Retail Predictive Application Server Administration Guide*.

3. Verify that none of the files failed during the upgrade; this can be determined based on the output of RSP Manager. For example, a successful output message would read:

```
Validation complete...
Files Checked: 106
Files Passed: 106
Files Failed: 0
```

The RPAS upgrade process is complete.

Domain Upgrade Process

After you have upgraded/patched RPAS server, you should upgrade any individual domains to be synchronized with that version. For information about upgrading domains, refer to the Domain Administration chapter in either the Classic Client or Fusion Client version of the *Oracle Retail Predictive Application Server Administration Guide*.

ODBC/JDBC Upgrade Process

This section describes how to save and migrate your existing ODBC/JDBC configurations to the new version. If you do not have any existing configurations to migrate, you can remove the old version and install the new one.

ODBC Server

The following sections describe how to update your existing ODBC server to the new version.

Upgrading from 13.0.x

On all platforms, the 13.0.x ODBC Server configurations are stored in `openrda.ini` and `oadrd.ini`.

1. Before upgrading `$RPAS_HOME`, save a copy of `openrda.ini` and `oadrd.ini` to a temporary location.
2. Remove the 13.0.x version of the ODBC Server. To do this on Windows, run `setup.exe` and choose the Remove option. On UNIX platforms, the **ODBC** directory should be automatically overlaid when you upgrade your `$RPAS_HOME`.
3. Install the new version (13.1.2 or later) of the ODBC Server. Use the following information to migrate the 13.0.x configuration to the new ODBC Server.

Migrating Server Configuration

Table 10–1 shows the mapping of the configurations between 13.0.x and the new version (13.1.2 or later) of the ODBC Server

Table 10–1 Mapping Server Configuration

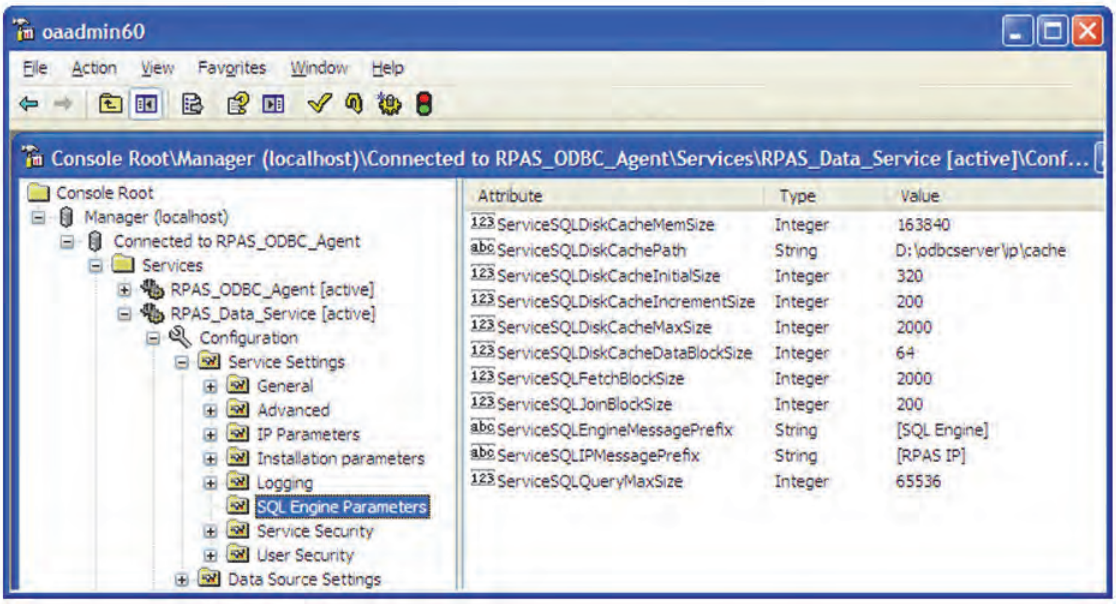
13.0.x Server Configurations	Corresponding Server Configurations (13.1.2 and Later)
INITIAL_SIZE	ServiceSQLDiskCacheInitialSize

Table 10–1 (Cont.) Mapping Server Configuration

13.0.x Server Configurations	Corresponding Server Configurations (13.1.2 and Later)
INCREMENT_SIZE	ServiceSQLDiskCacheIncrementSize
MAX_SIZE	ServiceSQLDiskCacheMaxSize
DATABLOCK_SIZE	ServiceSQLDiskCacheDataBlockSize
CacheMemSize	ServiceSQLDiskCacheMemSize
FETCHBLOCK_SIZE	ServiceSQLFetchBlockSize
JOINBLOCK_SIZE	ServiceSQLJoinBlockSize
QueryMaxSize	ServiceSQLQueryMaxSize

Figure 10–1 shows the new ODBC Manager with the configuration attributes that are listed in Table 10–1.

Figure 10–1 Server Configuration Attributes (Versions13.1.2 and Later)



Migrating Data Source Information

A data source in `oadrd.ini` looks like this sample:

```
ADDRESS=hostname.example.com
PORT=1710
CONNECT_STRING=/vol.nas/u08/Position_parent/croad_SR
TYPE=BTREE
SCHEMA_PATH=
REMARKS=
```

The attributes that you need to migrate are ADDRESS, PORT, and CONNECT_STRING. CONNECT_STRING in 13.0.x maps to DOMAIN_PATH in the new version. Figure 10–1 and Figure 10–2 show where they are in the new Server configuration.

Figure 10–2 Server Address and Port Number (Versions13.1.2 and Later)

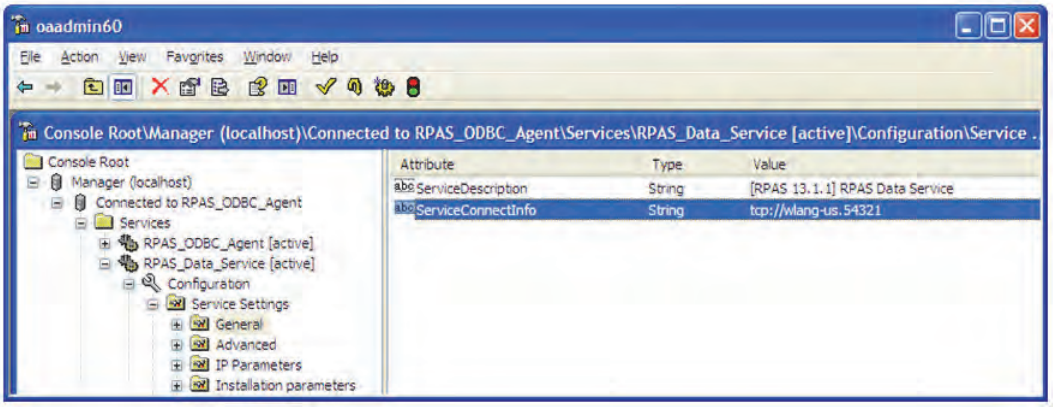
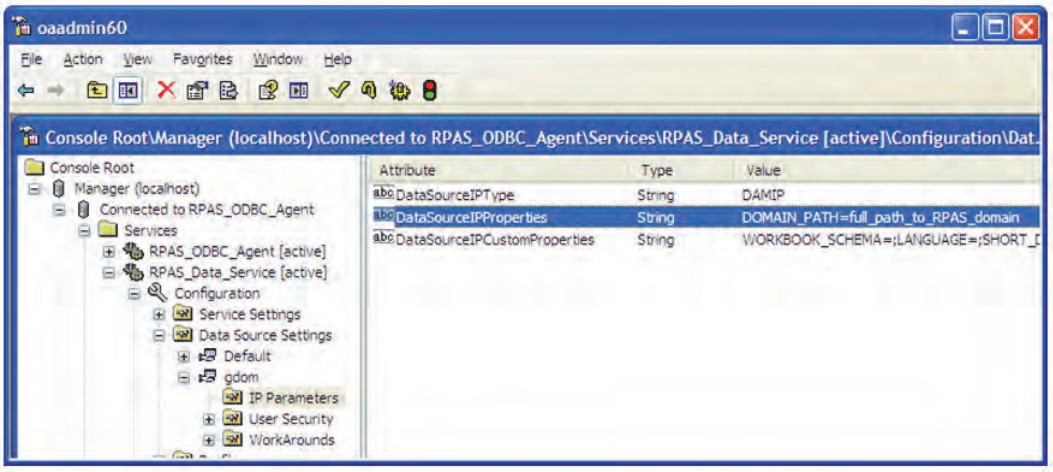


Figure 10–3 DOMAIN_PATH (Versions13.1.2 and Later)



Upgrading from 13.1.1.x

Perform the following procedure when upgrading from 13.1.1.x.

1. Before removing 13.1.1.x, take screenshots of the server configuration and server address as shown in these previous figures:

   - Figure 10–1, "Server Configuration Attributes (Versions13.1.2 and Later)"
   - Figure 10–2, "Server Address and Port Number (Versions13.1.2 and Later)"
   - Figure 10–3, "DOMAIN_PATH (Versions13.1.2 and Later)"

   These figures show the server's connecting and configuration information as well as the data sources you have.

2. Remove 13.1.1.x ODBC Server. To do this on Windows platform, run `setup.exe` in the server installation package and select the **Remove** option. On UNIX platforms, delete the **ODBC** directory under `$RPAS_HOME`.

3. Install the new version (13.1.2 or later) ODBC Server. Use the information saved in the screenshots created in Step 1 to complete the server and data source configuration.

ODBC Client

The following sections describe how to update your existing ODBC client to the new version.

UNIX Platform

In 13.0.x, the client configuration information to be migrated is stored in `oadrd.ini` and `odbc.ini`.

Note that `odbc.ini` is not required by the 13.0.x version of the RPAS ODBC Client, but it may be required by your ODBC application (such as OBIEE).

Example 10–1 Sample Data Source Definition in oadrd.ini

```
ADDRESS=hostname.example.com
PORT=54321
REMARKS=
```

To migrate SampleDataSource to the new version (13.1.2 or later) of the ODBC Client, create an entry for SampleDataSource in `odbc.ini`:

Example 10–2 Create an Entry for SampleDataSource in odbc.ini

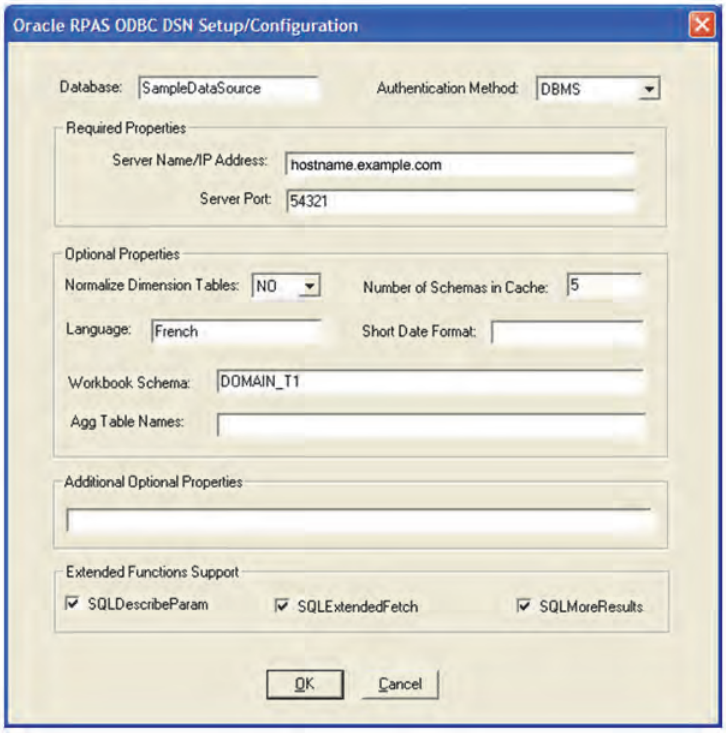
```
[ODBC Data Sources]
SampleDataSource=Oracle Retail RPAS ODBC Driver

[SampleDataSource]
Driver=PATH_TO_ODBC_CLIENT/odbcclient32/lib/ivoa22.so
Description=Oracle Retail RPAS ODBC Driver
Host=hostname.example.com
Port=54321
ServerDataSource=gdom
UseLDAP=0
DistinguishedName=
Encrypted=0
LoadBalancing=0
AlternateServers=
ConnectionRetryCount=0
ConnectionRetryDelay=3
CustomProperties=
```

The `customProperties` entry in `odbc.ini` can be copied to the same entry in the new version of `odbc.ini`.

Windows Platform

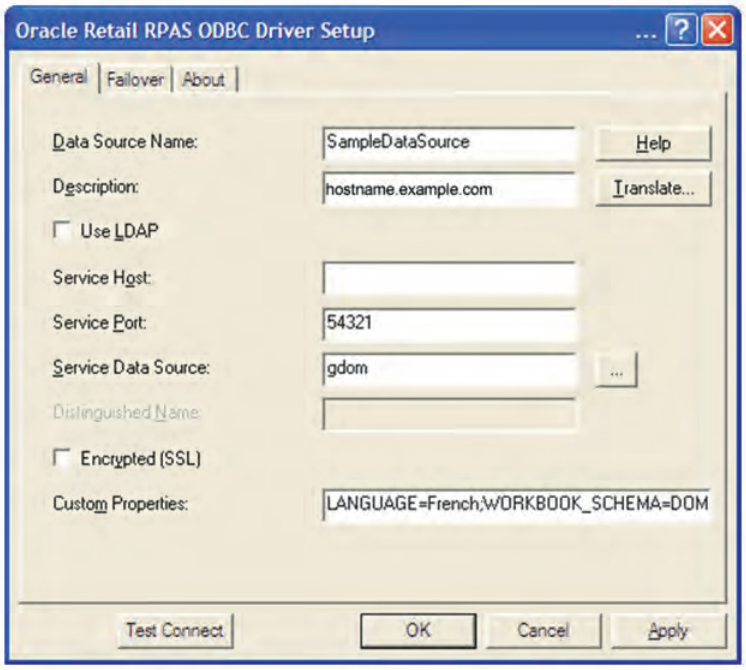
Figure 10–4 shows a sample data source in the 13.0.x version of the ODBC Client.

Figure 10–4 RPAS ODBC DSN Setup/Configuration for 13.0.x



In 13.3 and later versions, the Microsoft ODBC Administrator is used for creating and configuration the ODBC data source.

Figure 10–5 RPAS ODBC Driver Setup (Versions13.1.2 and Later)



All properties in Optional Properties and Additional Optional Properties in 13.0.x map to the Custom Properties in the new version (13.1.2 or later).

The properties are:.

- NORMALIZE_DIM_TABLES
- SCHEMA_IN_CACH
- LANGUAGE
- WORKBOOK_SCHEMA
- AGG_TABLE_NAMES
- SHORT_DATE_FORMAT
- DEFAULT_SCHEMA
- LOG_FILE
- RPAS_LOG_LEVEL

JDBC Client

The following sections describe how to update your existing JDBC client to the new version.

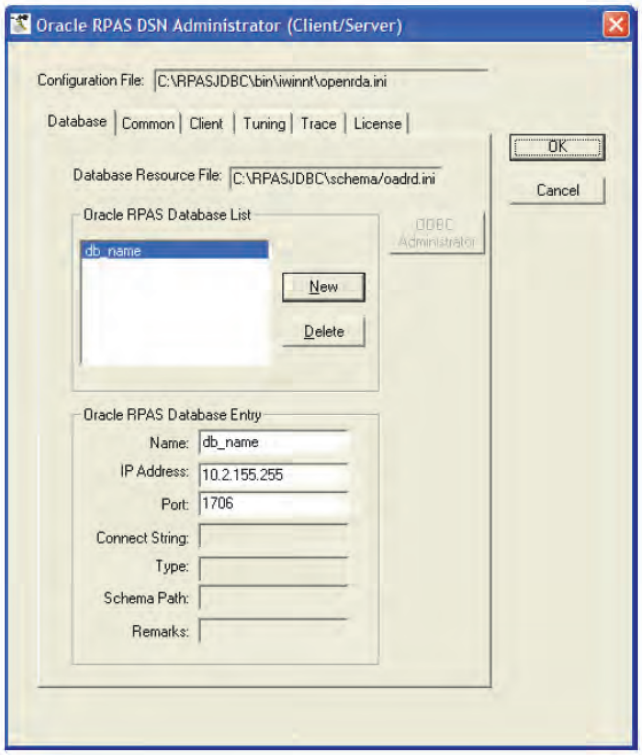
Upgrading from 13.0.x

Perform the following procedure when upgrading from 13.0.x

1. Before removing the 13.0.x JDBC Client, record the data sources that you want to migrate to the new version (13.1.2 or later).
   - To do this on Windows platforms, open the Admin Tool (shown in Step 3), gather the name, IP address, and port (of the server).
   - On UNIX platforms, use the command line Admin utility `rpasjdbcclientadmin` to gather the same information.

   You should also collect the optional custom connection properties that may exist in your JDBC URLs if you use URL in your JDBC applications.
2. Remove the 13.0.x JDBC Client.
   - To do this on Windows, run `setup.exe` and choose the **Remove** option.
   - On UNIX, delete the **jdbcclient** directory.
3. Install the new version (13.1.2 or later) of the JDBC Client. For instructions, refer to the RPAS ODBC/JDBC Driver chapter in either the Classic Client or Fusion Client version of the *Oracle Retail Predictive Application Server Administration Guide*. Then use the information gathered in Step 1 to construct the URLs for your JDBC applications.

Figure 10–6 RPAS DSN Administrator (Client/Server) Window



Upgrading from 13.1.1.x

Perform the following procedure when upgrading from 13.1.1.x.

1. Gather the information for the server's IP address, port number, data source name, and any custom connection properties in the 13.1.1.x JDBC URLs.
2. Remove the 13.1.1.x version of the JDBC Client.
3. Install the new version (13.1.2 or later) JDBC Client.
4. Use the information gathered in Step 1 to construct the JDBC URLs for the new JDBC Client.

11

RPAS Fusion Client Patch Installation

The Fusion Client patch process follows the same process as the full installation. For detailed instructions, refer to the section, "Installation Tasks" in Chapter 5, "Installing the RPAS Fusion Client". Before applying the patch, ensure that you backup the Fusion Client installation as a precaution.

When applying a Fusion Client patch, the existing configuration files are backed up and applied to the new installation. Ensure that the configuration files are restored correctly and that the `Foundation.xml` file is correct. For more information on the configuration files, refer to the *Oracle Retail Predictive Application Server Administration Guide for the Fusion Client*.

Note: After applying any Fusion Client patch, ensure that all users clear their browser cache.

The patch installation will create a `Foundation.xml` file based on the information provided in the installation properties file (`ant.install.properties`).

If the Solution ID is the same as the one already in the `Foundation.xml`, the patch installation will update the Solution ID parameters with the ones specified in the `ant.install.properties` file.

12

RPAS Classic Client Patch Installation

The RPAS Classic Client can be installed through either of the following methods:

- Windows Installer Method
- Classic Client Web Deployment

The following sections describe the installation processes for these two methods.

Windows Installer Method

This section describes the installation of the RPAS Classic Client on Windows machines, and describes how to configure the client to connect to a domain.

Make RPAS Classic Client Files Generally Accessible

Perform the following procedure to make the RPAS Classic Client available.

1. Create a directory on the network from where users will install the RPAS Classic Client.

   The location and the name of the directory are up to the system administrator's preferences. This directory is henceforth referred to as the **[RPASCLIENT]** directory.

2. Extract the client from the `ARPOPlatform-14.1.2.clients.zip` included in the main package to the **[RPASCLIENT]** directory.

Installing the RPAS Classic Client

The RPAS Classic Client installation procedure is the same for all of the RPAS applications. Perform the following procedures to install the application onto a PC.

1. Run the `setup.exe` file located in the **[RPASCLIENT]** directory on the network.
2. The welcome page displays. Follow the installation procedures as prompted.

   The setup program exits after the installation is complete.

Configuration

After creating an RPAS domain and starting the DomainDaemon (refer to the Classic Client or Fusion Client version of the *Oracle Retail Predictive Application Server Administration Guide*.), you must configure the RPAS Classic Client to connect to the domain on a server. This section provides instructions for configuring the RPAS Classic Client on a local computer using a Microsoft Windows operating system.

The EConfigure Utility

EConfigure is a Windows application that configures the client-server communication for RPAS. EConfigure lets you specify communication parameters and produces a file that is used as input to the client. These files must be in FCF (Foundation Configuration File) format/extension. The files contain the necessary information for the client to start up the communication with the server. These files can be stored on the client machine or on the network.

When the client is run, a file named `Foundation.FCF` is expected in the same directory. If the file has a different name or if it is stored somewhere on the network, the path to this file must be passed in as an argument to the client.

EConfigure consists of a menu bar, a main view, and the advanced settings dialog box. Passwords saved in the FCF file are encrypted. To launch EConfigure, double-click the `EConfigure.exe` file, which is by default located in the root directory of the RPAS Classic Client.

Menu Bar

The files produced by EConfigure may contain multiple connections. Each connection will be specific for a server with certain communication settings. Connections need to have unique descriptions, and they can be added and deleted using the menu bar.

Figure 12–1 EConfigure Menu Bar

Main View

The main view has the basic connection parameters. On this view, three groups of controls are available:

- The connection group
- The domains group
- The Advanced Settings dialog

The Connection Group

The connection group has these fields:

Field	Description
Database Server	The hostname or the IP address of the server, for example, atldev03 or 10.2.1.23. This value should be localhost when running the RPAS Server on a Windows machine.
Daemon Port	The port number on which the domain daemon is listening. This must be an integer between 1025 and 65535 (for example, 55278).

The Domains Group

The domains group has these fields:

Field	Description
Domain	The name of the domain that is displayed to the user when logging in. Select a domain from the list or type the name of a new domain and click **Add Domain**. You can delete a domain from the list by selecting it and then clicking **Delete Domain**.
Domain Path	The full path to the directory containing the domain, for example, **/root/testenv/domain/Sample_Project**.
User	Provide the user ID if you do not want to force the user to provide it when logging in. The user ID must be defined in the associated domain.
Password	Provide the password for the user if you do not want to force the user to provide it when logging in. This password must match the password defined in the domain for the associated user.

The Advanced Settings Dialog

The Advanced Settings Dialog has these fields for the default database login:

Field	Description
User	The database user that is used by the client if a domain specific user has not been entered, for example, adm.
Password	Like the default database user, default database password is used if a domain specific password has not been entered, for example, adm.
Database Port Range	Port range is used to specify the range of ports on which the RPAS Server processes is started by the DomainDaemon (the rpasDbServer processes). The port Start and port End fields are the lower and upper limits of this range respectively. These fields must be integers between 1025 and 65535, which are also the default values if values are not specified, for example, Start: 40000, End: 45000.
Compression Threshold	The number of bytes above which client and server are using compression. Only advanced users should manipulate this number.
Web Tunneling	The configuration of Web tunneling is no longer supported.
Proxy Settings	The configuration of the RPAS Classic Client to support a proxy server is not completed in this utility.

Classic Client Web Deployment

For instructions on Classic Client web deployment for patch installations, refer to Chapter 7, "RPAS Classic Client Web Deployment". If the web environment needs to upgrade, the web deployment instructions for full installations and patch installations are the same. If the web environment is unchanged, install the new WAR file after the same preparation.

13

HSA Patch Installation

An RPAS HSA system comprises of one or more integrated domains and the RPAS Data Mart (RDM). The integrated domains are upgraded and patched the same way as an independent domain. The RDM is upgraded and patched separately. All the domains do not need to be on the same version of RPAS, as long as these RPAS versions support the same schema version. RDM should be upgraded to the highest version among these RPAS versions and any RDM utilities must run on that version.

Note: For information on installing HSA, see Installing and Building the RPAS HSA Environment.

For more information about upgrading or patching RDM, refer to the Classic Client or Fusion Client version of the *Oracle Retail Predictive Application Server Administration Guide*.

Upgrading RDM

The RDM maintains a numeric schema version number which is independent of the RPAS version. This schema version number is incremented whenever there is a schema change making the schema incompatible with existing RPAS releases.

The `upgradeRdm` utility is used to upgrade the RDM schema and the RDM repository. This utility is responsible for updating the RDM with any RPAS revisions to the schema tables, PL/SQL procedures or the RDM repository directory structure. This utility must be run on the RDM each time a new version of RPAS is deployed. However, it will only increment the schema version when the changes are incompatible with existing versions of RPAS. This allows the integrated domains to run on different versions of RPAS as long as they support the same schema version.

Patching RDM

Similar to patching a domain when its configuration has been modified, in an HSA environment, the RDM can also be patched. The `patchRdm` utility applies a changed integration configuration to the RDM. The shared hierarchies, facts, and integration map components of an integration configuration can be patched.

The `patchRdm` utility takes the modified integration configuration as input and compares it against the existing integration configuration (a copy of the current integration configuration is always maintained in the RDM Repository directory area). If any differences are found, it updates the RDM schema. Once the `patchRdm` completes successfully, the integration configuration stored in the RDM Repository is replaced with the new one. Note that most of the components of the integration

configuration are patchable. However, removing a hierarchy and removing a dimension are not supported.

A

Appendix: Bandwidth Requirements

Understanding Bandwidth Requirements

The bandwidth requirements for a Web-based deployment of the RPAS Classic Client are minimal. The only large data transfer that occurs in this configuration is installation of the RPAS Classic Client to a PC (approximately 5 MB of data). This happens very infrequently. The client software is installed the first time a PC tries to connect to a domain or if the PC has an older version of the software that needs to be upgraded.

B

Appendix: RPAS Sizing and Partitioning Considerations

This appendix provides guidelines and information on what to consider when sizing and partitioning RPAS domains. This appendix is not specific to any one solution. It is meant to give general information that will help you size and partition your solutions to achieve optimal performance.

RPAS Sizing

The number of positions within the hierarchies of a solution has an effect on the on-line and batch performance of a domain. When using a global domain, the positions along the partitioned hierarchy will be split among local domains. This partitioning will help in certain areas but is not a reason to include large numbers of positions in a single global domain environment.While there is no hard limit on how big a single global domain environment should be, the number of positions within the lowest level of each hierarchy should not be excessive. There are certain batch operations (loading hierarchies, reshaping arrays, repartitioning data between domains) that will be affected no matter how many local domains are created.

For example, assume that there is a solution that has a product, location and calendar hierarchy. In one environment, you have a single global domain instance with the product hierarchy having 1 million positions at the lowest level, the location hierarchy having 100 positions and the calendar hierarchy having 5 years. In a second environment, you have two global domain instances each with 500,000 product positions, 100 locations and 5 years of data. The loading of the product hierarchy in the first environment will be longer in than the second environment no matter how the local domains are partitioned.

Partitioning Considerations

The purpose of using a global domain and partitioning data across multiple domains is to help reduce contention, provide smaller domains for most users to interact with and to allow for parallel processing during batch. If the partitioning is not done correctly, it can lead to unnecessary contention or poor performance. Here are some key considerations to make when determining how to partition a global domain environment.

- Try to keep the database file sizes under 2GB in each local domain. There is no restriction as such to this hard limit in the current RPAS embedded b-tree database, but as a rule of thumb it can be followed to get better performance.
- Try to limit to one measure per database in the configuration as it will help in reducing the database contention.
- The hierarchy that you partition on should allow the users the ability to work in a single local domain. If users require access to all positions within a hierarchy, that is not a good candidate for partitioning. For example, it does not make sense to partition on the location hierarchy if your business process requires all users to include all locations in each workbook. Unless dictated by business needs, try to avoid partitioning on alternate hierarchy and do the partitioning on the main hierarchy.
- The partition level should also be higher than the level at which most of the data is stored. If most data is stored at the division level or below in the product hierarchy, the partition level should be at the division level or above. When data is based above the partition level of the domain, the data will be stored in the master domain. All users across the local domains who require this data will have contention from all other users and not just from the users of the local domains they are currently working with. Try to have as few users as possible to work on the master domain based on the hierarchy levels at which the different user roles operate. Try to have majority of the users to work below the partition level and evenly distributed across local domains.
- The partitioning should be set such that the business requirements do not require high usage of the master domain. The performance of a workbook built from the master domain will never match that of a local domain workbook. The heavy usage of workbooks should take place across the local domains. For example, if most of the users only need to see data within a division then the partitioning should not be done below that level. Try to adopt a partitioning strategy which can result in majority of the measures to be at lower base intersection (measures whose intersection is at or below the partitioning dimension) and which can result in having one partition level position per local domain for improved performance and reduced contentions.
- The number of users that are in a single local domain should be evenly distributed across all the domains in a global domain environment based on their areas of responsibility. If there are a larger number of users in a single local domain than others, it will not matter how many partitions you create. The domain with the largest user group will always have the potential to experience more contention issues and poor performance. If possible, create more domains and separate more users across those domains.
- For very large global domains, we may have the local domains reside outside of the master domain so that they can be distributed amongst multiple file systems for better performance.

Workbook Sizing Considerations

The impact of size for the end user is not limited to only the size of the domain or where they are building a workbook from. The size of the individual workbooks will have a direct affect on the performance they experience. The workbook size is a result of the number of measures and number of positions from each of the hierarchies included in the built workbook.

> **Note:** The Fusion Client user interface hides the distinction between local and global domains and implicitly establishes user interaction with the domains to which the user has access. A user does not have to explicitly select a domain (through a profile selection during login) to work in the RPAS Fusion Client. For additional information, refer to the *Oracle Retail Predictive Application Server User Guide for the RPAS Fusion Client*.

The number of measures for a workbook template is static based on what is configured. The more measures that are configured in a template the larger the workbook becomes. As workbooks get larger, workbook operations will take longer. Specifically, operations like save and open are directly related to the overall size of the workbook.

Since the number of measures in a given workbook template are static based on what is configured, the number of positions in each hierarchy is the only factor that the end user controls from workbook to workbook using the same template. Two workbooks for the same template may have completely different performance based on how many positions are included.

The simplest way to compare the size of two workbooks for the same template is to multiply the number of positions for each hierarchy at the base intersection of the template and the measures. For example, assume that there is a workbook that has the majority of measures based at the week/style-color/channel. This workbook always contains 500 measures so that is a constant. If there is one workbook that contains 52 weeks (One year), 300 style colors and 3 channels, the total possible positions at the base level would be slightly over 23 million cells. This does not include any aggregate values a user may view. If a user built the same workbook for two years (104 weeks), the total possible positions double to over 46 million cells. Going back to the first example and include 450 style colors instead of 300, the total possible base level cells would increase to over 35 million.

Although there is no maximum number of cells that should be contained in a workbook, the number does have an impact on performance and therefore should be considered during design. If workbooks contain a total possible number of positions at the base level in the hundreds of millions, not only will the workbook performance be less than ideal but also the user will not be able to manage that level of detail.

C

Appendix: rsp_manager Usage

The `rsp_manager` (Retail Service Pack Manager) is a Perl script tool that is capable of currently patching the following:

- RPAS
- Configuration Tools
- Domains
- Solution Environments (AIP_HOME, SCI_HOME, and so on)

This system will automatically run any creates, scripts, or procedures that need to be run when a patch is applied.

Prerequisites

You must have Perl 5.005 or later installed on your system. Most UNIX variants will have this installed by default. On Windows, an installation of Cygwin is required.

Applying a Service Pack

The following instructions describe how to install a service pack.

1. Copy component`-ver.os.tar.zip`

   Note: This component will be named `-ver.os.zip` for Windows.

   The service pack is shipped as a compressed `.tar` or `.zip`, depending on the platform. The file will be named based on the release level of the code contained within. Copy the service pack `.tar` or `.zip` file to a standard service pack directory that you have previously set up.

Example C–1 Service Pack Directory

Create a service pack directory at:

/files1/service_packs

Then, copy the service pack file to (where **/service_packs** is a user-created directory for archived service packs):

cp ./ARPOplatform-14.1.2.sun.tar.zip /service_packs

In Windows, this directory would resemble the directory **C:\service_packs**.

2. Unpack the service pack from the `.tar` or `.zip` file from Step 1.

Example C–2 UNIX Commands

The following are example UNIX commands to unpack the service pack:

```
cd /service_packs

unzip ARPOplatform-14.1.2.sun.tar.zip

tar -xvf
```

This creates a subdirectory in your service pack directory named the same as the service patch/pack version, which contains a directory for the platform. In this directory, you will see subdirectories for each of the modules this service pack is updating.

For example, if this service pack has updates to RPAS, domains, and tools, the following module directories could be created:

/service_packs/ARPOplatform/14.1.2/sun/rpas

/service_packs/ARPOplatform/14.1.2/sun/domain

/service_packs/ARPOplatform/14.1.2/sun/tools

Additionally, this document, Release Notes, and a copy of any service pack installation scripts/libraries will be copied to a location such as:

```
/service_packs/ARPOplatform/14.1.2
```

Note: In Windows, you may use WinZip or a similar unzipping tool for this step.

---

3. Apply the service pack on a staging or production server. After extracting the service pack, you will have to apply the service pack to the installed components and any domains that have been built. Note that the service pack installation should be carried out on the same operating system as that on which the product resides.

   The syntax of applying a service pack with *rsp_manager* is:

   ```
   rsp_manager –install –sp <sp path> -domain <domain path>
   ```

Example C–3 Commands to Apply Service Pack 14.1.2 for ARPO platform and domain

To apply service pack 14.1.2 for ARPO platform and domain /domain1, use the following commands:

```
cd /service_packs/ARPOplatform/14.1.2/

./rsp_manager –install –sp sun –domain /domain1
```

Example C–4 Commands to Turn on File Logging

To turn on file logging of the output and store the results of the application in `domain1.log`, use the following commands:

```
cd /service_packs/ARPOplatform/14.1.2/
```

```
./rsp_manager –install –sp sun –domain /domain1 –log domain1.log
```

Following installation, a validation process will be run against your patched install.

Applying Service Packs on Multiple Domains

If you have more than one domain running off the same ARPOplatform, it is possible to create a domain list file and supply that file path, instead of the domain path, as an argument. This will be a text file with a full path to a domain on each line.

The additional syntax of `rsp_manager` is:

```
rsp_manager -install -sp <sp path> -domain <domain_list_file>
```

Example C–5 Applying Service Packs

To install service pack 14.1.2 for ARPOplatform and all domains listed in `/files/domain_list.txt`, use the following commands:

```
cd /service_packs/ARPOplatform/14.1.2/
```

```
./rsp_manager -install -sp sun -domain /files/domain_list.txt
```

…where `/files/domain_list.txt` looks similar to this:

```
cat /files/domain_list.txt
```

```
/domains/domain1
```

```
/domains/domain2
```

```
/domains/domain3
```

Note: Logging will work with multiple domains, but only one file will be created and written to. This single log will contain the output from all updated modules and domains.

Optional Arguments or Commands for rsp_manager

The `rsp_manager` has optional arguments that allows you to perform tasks such as an installation version report, install or patch validation, and patch application forcing. The following sections provide details on these arguments.

-no_rpas

This flag stops your **RPAS_HOME** from being patched. This can be used in conjunction with `-no_tools`, but still passing domains, in order to upgrade a domain to your **RPAS_HOME** code level without performing an RPAS upgrade.

-no_tools

This flag stops your RIDE_HOME from being patched. This can be used in conjunction with `-no_rpas`, but still passing domains, in order to upgrade a domain to your **RPAS_HOME** code level without performing a tools upgrade.

-no_domain

This flag stops the patch from being applied to any domains.

-log <logfile>

Although logging is done by default, the log will be saved to the current working directory as a date-stamped filename. This flag allows you to change the name of the log file created to the argument you pass.

-force

This flag enforces the application of the patch/pack regardless of the versions that the components report. This allows you to reapply a patch that has already been applied, while also running any update scripts that might have already been run.

Example C–6 -force Flag

To force reinstallation of the 14.1.2 service patch onto your installation and a single domain, with logging:

```
cd /service_packs/ARPOplatform/14.1.2/

./rsp_manager -install -sp sun -domain /domain1 -log domain1.log -force
```

-validate

This flag is run by default at the end of all `-install` processes. The purpose of this flag is to validate components of the service patch/pack against your installation. Currently, this will only check your core binaries and libraries in both **RPAS_HOME** and RIDE_HOME. Tests include permissions comparisons and file checksum validation, which are represented in the output as ***P*** or ***C*** respectively upon errors.

Example C–7 -validate Flag

To validate the 14.1.2 Sun service patch against your installation any time after patching:

```
cd /service_packs/ARPOplatform/14.1.2/

./rsp_manager -validate -sp sun

Validating your updated install against the service pack/patch -

.....................................................

Validation complete...

Files Checked: 106

Files Passed: 106

Files Failed: 0
```

Note: The number of files checked might not match the previous number, as it changes quantity based on platform and patch version. If any files fail, a listing of those failed files will be presented, each being preceded by flags (***C*** or ***P***) to indicate which checks failed.

Optional Environment Variables

Note: The following process is not a suggested installation process, but Oracle Retail acknowledges that it may be more efficient in some limited cases.

Since rsp_manager relies on the Retek.pm library, this file must normally be in the same directory as the one you run rsp_manager from. You may use an environment variable to point to the path that contains the library so that they can be split from each other. A useful instance would be setting up a directory in your path and placing rsp_manager inside, while using the RSP_HOME environment variable to point to the directory that contains the Retek.pm file. This would allow you to run rsp_manager from anywhere on the system.

Example C–8 Running rsp_manager

To be able to run rsp_manager from anywhere:

```
> cd /service_packs/ARPOplatform/14.1.2/

> ls

rsp_managerRetek.pm

> mkdir ~/bin

> cp ./rsp_manager ~/bin/

> export PATH=~/bin:$PATH

> export RSP_HOME=/service_packs/ARPOplatform/14.1.2
```

At this point, you can change directories to anywhere on the disk and run rsp_manager.

Keep in mind that if you do choose to split these files, when you obtain new copies of the script and library, you will need to place them into the locations you reference in $PATH and $RSP_HOME.

D

Appendix: Oracle Single Sign-On (SSO)

This appendix provides a basic description of Oracle Single Sign-On (SSO) and addresses these topics:

- What is Single Sign-On?
- What Do I Need for Oracle Single Sign-On?
- Oracle SSO Terms and Definitions
- What SSO is Not
- Single Sign-On Topology
- Installation Overview
- User Management

What is Single Sign-On?

SSO is a term for the ability to sign onto multiple Web applications through a single user ID/Password. There are many implementations of SSO. Oracle currently provides three different implementations:

- Oracle SSO (OSSO)
- Java SSO (with the 10.1.3.1 release of OC4J)
- Oracle Access Manager (OAM 11g)

> **Note:** OAM 11g provides more comprehensive user access capabilities. For additional support information, refer to Hardware and Software Requirements.

Most, if not all, SSO technologies use a session cookie to hold encrypted data passed to each application. The SSO infrastructure has the responsibility to validate these cookies and, possibly, update this information. The user is directed to log on only if the cookie is not present or has become invalid. These session cookies are restricted to a single browser session and are never written to a file.

Another facet of SSO is how these technologies redirect a user's Web browser to various servlets. The SSO implementation determines when and where these redirects occur and what the final window shown to the user is.

Most SSO implementations are performed in an application's infrastructure and not in the application logic itself. Applications that leverage infrastructure managed

authentication (such as deploying specifying Basic or Form authentication) typically have little or no code changes when adapted to work in an SSO environment.

What Do I Need for Oracle Single Sign-On?

The nexus of an Oracle SSO system is the Oracle Identity Management Infrastructure installation. This consists of the following components:

- An Oracle Internet Directory (OID) LDAP server, used to store user, role, security, and other information. OID uses an Oracle database as the back-end storage of this information.
- An Oracle HTTP Server 11g Release 1 as a front end to the Oracle WebLogic Server. The Oracle HTTP Server is included in the Oracle Web Tier Utilities 11g Release 1 (11.1.1).
- An Oracle SSO Plug-in (OAM 11g WebGate) is used to authenticate the user and create the OSSO session cookie. This is available in the Oracle Fusion Middleware 11g Identity and Access management 11g package.
- The users and group information may also be loaded or modified through standard LDAP Data Interchange Format (LDIF) scripts.
- Additional administrative scripts for configuring the OSSO system and registering HTTP servers.

For more information on setting up SSO, refer to either the Classic Client or Fusion Client version of the *Oracle Retail Predictive Application Server Administration Guide.*

Additional WebLogic managed servers are needed to deploy the business applications leveraging the OSSO technology.

Oracle SSO Terms and Definitions

This section lists the terms and definitions used in Oracle SSO.

Authentication

Authentication is the process of establishing a user's identity. There are many types of authentication. The most common authentication process involves a user ID and password.

Identity Management Infrastructure

The Identity Management Infrastructure is the collection of product and services which provide Oracle SSO functionality. This includes the Oracle Internet Directory, an Oracle HTTP server, and the Oracle SSO services. The Oracle Application Server deployed with these components is typically referred as the Infrastructure instance.

mod_wl_ohs

`mod_wl_ohs` operates as a module within the HTTP server that allows requests to be proxied from the Apache HTTP server to the WebLogic server.

Oracle Internet Directory

Oracle Internet Directory (OID) is an LDAP-compliant directory service. It contains user IDs, passwords, group membership, privileges, and other attributes for users who are authenticated using Oracle SSO.

Partner Application

A partner application is an application that delegates authentication to the Oracle Identity Management Infrastructure. One such partner application is the Oracle HTTP Server (OHS) supplied with the Oracle Application Server. OHS uses the OAM 11g WebGate module to configure this functionality.

All partner applications must be registered with the Oracle Access Manager.

Realm

A Realm is a collection of users and groups (roles) managed by a single password policy. This policy controls what may be used for authentication (for example, passwords, X.509 certificates, and biometric devices). A Realm also contains an authorization policy used for controlling access to applications or resources used by one or more applications.

A single OID can contain multiple Realms. This feature can consolidate security for retailers with multiple banners or to consolidate security for multiple development and test environments.

What SSO is Not

SSO is not a user ID/password mapping technology.

However, some applications can store and retrieve user IDs and passwords for non-SSO applications within an OID LDAP server. An example of this is the Oracle Forms Web Application framework, which maps OSSO user IDs to a database logins on a per-application basis.

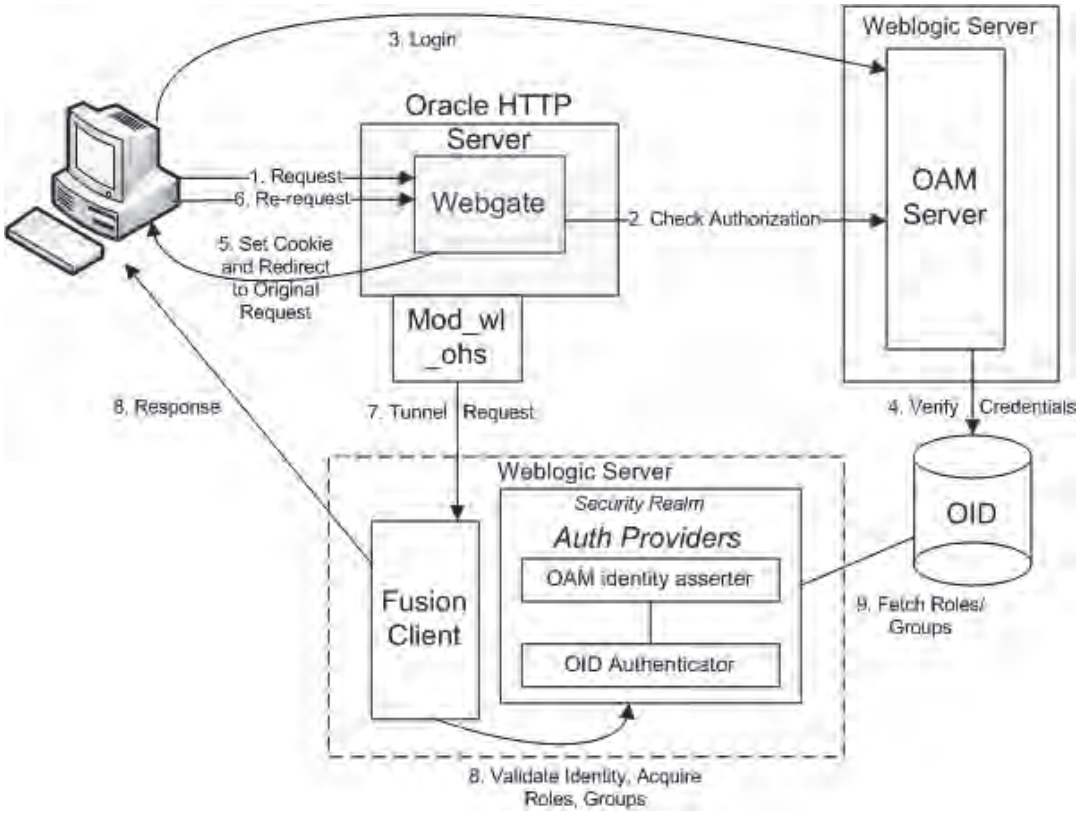
Single Sign-On Topology

RPAS usernames are case sensitive. Therefore, when setting up an SSO environment, ensure that the case sensitivity is maintained.

Note: For additional information, refer to the chapter, "Configuring Single Sign-On with Oracle Access Manager 11g" in the *Oracle Fusion Middleware Application Security Guide*.

Figure D–1 illustrates the SSO topology.

Figure D–1 Single Sign-On Topology



Installation Overview

For Oracle SSO installation, refer to the *Oracle Fusion Middleware Application Security Guide* 11g Release 1 (11.1.1)Part Number E10043-09.

The section, Deploying the Oracle Access Manager 11g SSO Solution (http://docs.oracle.com/cd/E21764_01/core.1111/e10043/osso_b_oam11g.htm#BABHHABA), provides information on how to implement OAM 11g with the Authentication Provider when you have applications that are (or will be) deployed in a WebLogic container.

User Management

User Management consists of displaying, creating, updating or removing user information. There are two basic methods of performing user management: LDIF scripts and the Delegate Administration Services (DAS) application.

OID DAS

The DAS application is a Web-based application designed for both administrators and users. A user may update their password, change their telephone number of record, or modify other user information. Users may search for other users based on partial strings of the user's name or ID. An administrator may create new users, unlock passwords, or delete users.

The DAS application is fully customizable. Administrators may define what user attributes are required, optional or even prompted for when a new user is created.

Furthermore, the DAS application is secure. Administrators may also what user attributes are displayed to other users. Administration is based on permission grants, so different users may have different capabilities for user management based on their roles within their organization.

LDIF Scripts

Script based user management can be used to synchronize data between multiple LDAP servers. The standard format for these scripts is the LDAP Data Interchange Format (LDIF). OID supports LDIF script for importing and exporting user information. LDIF scripts may also be used for bulk user load operations.

User Data Synchronization

The user store for Oracle SSO resides within the Oracle Internet Directory (OID) LDAP server. Oracle Retail applications may require additional information attached to a user name for application-specific purposes and may be stored in an application-specific database. Currently, there are no Oracle Retail tools for synchronizing changes in OID stored information with application-specific user stores. Implementers should plan appropriate time and resources for this process. Oracle Retail strongly suggests that you configure any Oracle Retail application using an LDAP for its user store to point to the same OID server used with Oracle SSO.

E

Appendix: Installing Solution Plug-ins

This appendix explains how to install and set up the Solution Plug-ins, along with the required configuration in the Fusion Client.

Solution Plug-ins are small external applications within the user interface of the Fusion Client. Such applications are termed as RPAS extensions or plug-ins. They are external in the sense that they present a user interface (UI) to interact with external systems that connect to RPAS or non-RPAS backend, not necessarily as Fusion Client Workbooks.

Assumptions

These instructions assume knowledge of application servers, databases, and application installation or administration, and are intended for system administrators and experienced IT personnel. Before carrying out any of these activities, ensure that you understand UNIX commands (including shell configuration and scripting), directory operations, and symbolic links.

In order to install the Solution Plug-in Purchase Order View (POView) along with an RPAS application like MFP, you must perform the following tasks listed in the section, Road Map Tasks.

Road Map for Installing the Solution Plug-ins

The Road Map for Installing the Solution Plug-ins details these topics:

- Installing Solution Plug-in
- Installing POView
- Road Map Tasks

Installing Solution Plug-in

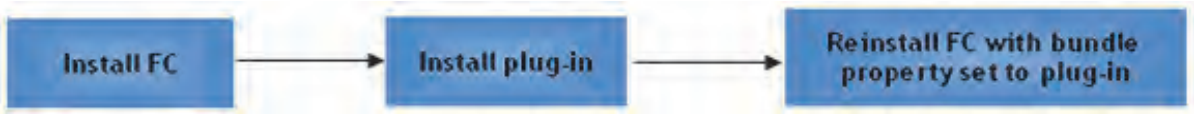
At a high level plug-ins are integrated with the Fusion Client by performing the following steps.

1. The Fusion Client's installer is run.
2. The plug-in bundle's installer is run. This installs a shared library in the WebLogic Server.
3. The Fusion Client's installer is re-run with bundle property set to the plug-in name. This creates a reference to the above-mentioned shared library within the Fusion Client.

   Postinstalllation, configure the Taskflow_MultiSolution.xml and MultiSolutionBundle.properties as per the Solution Plug-in requirement. For

details, refer to the *Oracle Retail Predictive Application Server Configuration Tools User Guide*.

Figure E–1 Install Sequence for a Solution Plug-in



Installing POView

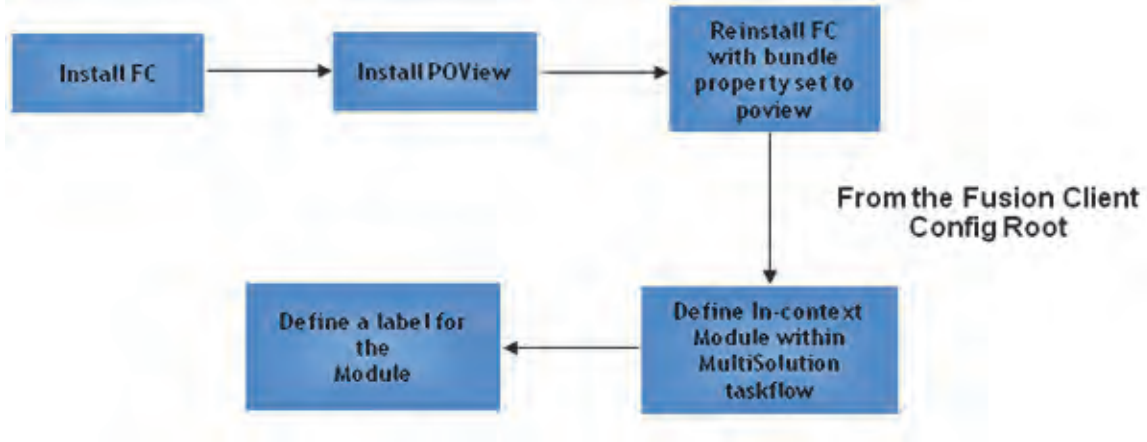
About POView

POView is a Solution Plug-in that can be used with RPAS applications like Merchandise Financial Planning (MFP). It provides users with an interface to look up the currently open or partially received purchase orders in RMS based on the product, location and time intersection of your selection. The selection contexts for POView are configurable through the manifest file. With this knowledge an MFP user can better plan their Open to Buy (OTB) budgeting and enhance their Merchandise plans.

POView Configuration

After completing the installation, the Fusion Client needs to be configured to display the POView plug-in within the proper UI context. The overall setup process is show in Figure E–2.

Figure E–2 Install and Configuration Sequence for POView



The POView setup process consists of the following steps.

1. Install Fusion Client using the instructions available in the Installation Guide. When initially installing Fusion Client make sure the input.installed.bundles property is set to blank, since POView has not been installed yet.
2. Install POView which works similar to Fusion Client Installer.
3. Re-run Fusion Client installer with the input.installed.bundles property set to poview.

4. Configure POView as in-context modules in Taskflow_MultiSolution.xml and MultiSolutionBundle.properties from the Fusion Client configuration root.

About the POView Installer

The POView Installation media includes an Oracle installer that you must run to install POView. The installer installs the application based on the parameters specified in an installation properties file.

You can install the application in either of these modes:

Silent Mode

Silent mode is non-interactive. In silent mode, the installer processes the values set in the properties file with no manual intervention required. Refer to Running the POView Installer—Silent Mode.

Graphical or Text Mode

In graphical or text mode, the Oracle Installer prompts you to enter or modify the value of properties specified in the installation properties file.Refer to Running the POView Installer—Graphical or Text Mode.

Road Map Tasks

The following sections provide details of each step of the setup process.

To install a Solution Plug-in, there are three phases and tasks within each phase as listed in Table E–1.

Table E–1 Road Map Phases and Tasks

Phase	Task	Additional Information
Pre-Installation Tasks	Installing the Fusion Client	For more information, refer to Installing the Fusion Client.
	Set up the install.properties file	For more information, refer to Setting Up Your Installation Properties File.
	Set the environment variables	For more information, refer to Setting up Environment Variables.
	Create credentials in the Oracle Wallet	For more information, refer to Creating User Credentials in an Oracle Wallet.
Installation Tasks	Install POView in silent mode or text/graphical mode.	For more information, refer to Running the POView Installer—Silent Mode and Running the POView Installer—Graphical or Text Mode.
Post-Installation Tasks	Installing the Fusion Client	For more information, refer to Fusion Client Install.
	Set up In-context module in Multi Solution Taskflow	For more information, refer to Configuring In-Context Module.
	Configure Manifest file	For more information, refer to Configuring the Manifest File.

Pre-Installation Tasks

Before starting POView for production, perform these tasks in order:

1. Installing the Fusion Client
2. Setting Up Your Installation Properties File
3. Setting up Environment Variables
4. Creating User Credentials in an Oracle Wallet

Installing the Fusion Client

For details on Fusion Client setup refer to Chapter 5, "Installing the RPAS Fusion Client" and any Oracle Retail product specific Installation Guide for the Fusion Client.

Note: Edit the `ant.install.properties` file using any text editor and `set input.installed.bundles` and `input.installed.bundles.shared.libs properties` to empty.

Setting Up Your Installation Properties File

Ensure that all previous pre-installation tasks are complete.

To install POview, it is recommended that you set up the installation properties file (ant.install.properties) before running the installer.

Note: Before installing the POView in silent mode, you must set up the installation properties file. This step is optional when installing in text or graphical mode.

To set up your install.properties file, perform the following steps:

1. Navigate to the POview directory, use an unzip utility to unzip `POView.zip`.
2. Copy the ant.install.properties.sample file to the same directory, and rename it ant.install.properties.
3. Edit the ant.install.properties file using any text editor, specifying values as described within the file, and save it. For more information on the parameters, refer to Installation Properties File Parameter Reference.

Note: Ensure that the ant.install.properties file is available in the same directory with the `install.sh/install.cmd` script.

Installation Properties File Parameter Reference

Table E–2 describes the parameters in the ant.install.properties file that you must set up before you install the POview application in silent mode:

Table E–2 Installation Properties File Parameters

Parameter Name	Description
Retrieve Credentials	
input.retrieve.credentials	This applies to GUI mode of install. For silent, you must set it to **Yes**. You should set the value to **Yes** if you have stored user credentials to retrieve them later. If you do not have user credentials stored in the secure wallet, or if you want to overwrite the existing credentials or store new credentials, set the value to **No**. If using the silent mode installer, then you must set `input.retrieve.credentials` to **Yes** and prepare the Wallet with WebLogic and Database credentials, before installing.
Wallet Directory	
input.wallet.dir	Specify the location of the Oracle Wallet where the user credentials are to be saved or retrieved.
WebLogic Admin Server Information	
input.wls.target	Specify the host name where the application server is running. If you need to specify a local host because one of your target managed server is on the local host (meaning that it is the same as your admin server host), enter any of the following: ■ localhost ■ 127.0.0.1 ■ The DNS name If the application server is running on a different host as the installer, enter the DNS name or IP address of the host where the application server is running.
input.admin.port	Specify the port number associated with the application server.
input.admin.user	Specify the administrative user name for the application server.
input.admin.username.alias	Specify an alias name for the administrative user. Specifying an alias name for the administrative user enhances the security for the application. When left blank, the alias name will default to the administrative user name.
Application Configuration Information	
input.is.multiple.hosts	To specify that your domain contains at least one managed server machine that is different from the admin server machine, set the value to yes. If all the managed server machines are on the same machine as the admin server machine, then set the value to no. In other words, even if you have a cluster with multiple managed servers or have multiple standalone managed servers, the value should be no if all of these managed servers are on the same machine as the admin server machine.
Application Server Information	
input.target.name	Specify the cluster or managed application server names where you want to install POView.
Application Deployment Details	
input.solution.id	Specify the Solution details for POView.
input.app.name.customlib	Specify the application deployment name.
Data Source Details	
input.datasource.url	Specify the Datasource URL in the format: jdbc:oracle:thin:@[host]:[port]:[dbname]

Table E–2 (Cont.) Installation Properties File Parameters

Parameter Name	Description
input.datasource.dsname	Specify the Datasource name to be created in the Application Server - PoViewDBDS
input.datasource.jndiname	Specify the JNDI name jndi/PoViewDBDS
input.datasource.alias	Specify a DataSource alias name
input.datasource.sid	Specify the System ID (SID) for the Database
input.run.sql	Specify this property value to false if you do not want to run the sql script for creating POView related Database objects during installation. If this property is set to false, then post installation user needs to manually create the DB Objects on the target database.
RPAS Fusion Client	
fusion.configpath	Specify the Fusion Client Configuration Directory. The directory where Fusion Client was Installed.
input.targetMachines	Specify the hostname/IP of the machine where Fusion Client is Installed.

Setting up Environment Variables

Before you start the installation, ensure that the following environment variables are set in the system:

- WEBLOGIC_DOMAIN_HOME

  Location where the WebLogic domain is installed.

- ORAINST_HOME

  An absolute path to the file containing the central inventory information. This file must define the inventory_loc and inst_group variables in it. The inst_group value should be the UNIX group for the opatch utility.

Example E–1 Environment Variables

```
inventory_loc=/home/weblogic/oraInventory
inst_group=users
```

> **Note:** Although it is recommended that these variables be set up in relevant bash shell startup files (`.bash_profile`) of the system, you can also set up the variables using the EXPORT command at the UNIX prompt. For more information on setting up these variables in the startup files, refer to the *Oracle Retail Advanced Inventory Planning Implementation Guide*.

Setting Environment Variables for the Current Session - UNIX

To set up the environment variables for the current session, at the UNIX prompt type the following commands in sequence:

```
export WEBLOGIC_DOMAIN_HOME=<path where the WebLogic domain is installed>
```

Example E–2 UNIX Command

```
/u01/app/oracle/middleware/user_projects/domains/base_domain
```

Setting Environment Variables for the Current Session - Windows

To set up the environment variables for the current session on Windows, at a command prompt type the following commands in sequence:

```
set WEBLOGIC_DOMAIN_HOME=C:\Oracle\middleware_wls1036\user_
projects\domains\base_domain
```

Creating User Credentials in an Oracle Wallet

Ensure that all previous pre-installation tasks are complete.

Information such as user credentials for the POView installation is encrypted and stored in a secure location in the application installation directory. This location is called the Oracle Wallet.

When the installation starts, the administrative user credentials are retrieved from the Oracle Wallet based on the alias name specified in this window.

Note: Before installing the POView in silent mode, you must create user credentials in an Oracle Wallet. This step is optional when installing in text or graphical mode.

Storing Credentials

Installing the POView in silent mode requires two kinds of credentials to be stored in the wallet before it starts:

- WebLogic Admin UserID and Password

  Stored against the alias wlsalias (or whatever alias is configured within the ant.install.properties file against the input.admin.username.alias property).

- Database UserID and Password

  Stored against the alias dsAlias (or whatever alias is configured within the ant.install.properties file against the input.datasource.alias property)

Note: The respective passwords are prompted for by the `save_credential.sh` script

---

Creating Credentials

To create the user credentials, perform the following steps:

1. Set and export the JAVA_HOME variable.
2. CD to `<installer>/retail-public-security-api/bin`.
3. Run `sh save_credential.sh <userNameAlias> <username> <locationOfWalletDir>`.

   The following table describes the field options to run the script:

Field Option	Description
<userNameAlias>	The keyname for which the credentials need to be stored.
<username>	The username to be stored in a secure credential wallet for the specified userNameAlias.

Field Option	Description
<locationOfWalletDir>	The directory where the wallet will be created. This is an optional parameter. If omitted, it creates the wallet under: `<installer>/retail-public-security-api/secure-credential-wallet`.

Note: Credentials can be used in GUI mode if you select **Yes** to Retrieve Credentials from the wallet as shown in Figure E–4, "Retrieve Credentials? Window".

Installation Tasks

To install the POView in silent mode, perform the following steps:

Running the POView Installer—Silent Mode

To install the POView in silent mode, perform the following steps:

1. Ensure that you have completed the task, Setting Up Your Installation Properties File.

   Note: Ensure that the ant.install.properties file is available in the same directory with the install.sh script.

2. Ensure that you have saved these user credentials in an Oracle Wallet:
   - WebLogic domain admin user credential (the entry alias is configurable)
   - Database user credential

   The silent mode installer does not ask for user credentials. Instead it retrieves them from the Oracle Wallet.

   For more information, refer to Creating User Credentials in an Oracle Wallet.

   Note: The installation property input.retrieve.credentials must be set to **Yes**.

3. Ensure that the WebLogic Server is running and the Database is accessible.
4. Navigate to the POView installation folder, enter the following command:

   ```
   ./install.sh silent
   ```

Running the POView Installer—Graphical or Text Mode

If you prefer to use GUI (graphical user interface), you can use the Oracle Installer in the graphical or text mode. Although this section describes how you can install POView in graphical mode, the same on-screen instructions appear as text instructions in the text mode.

Note: Although you do not have to set up the installation properties file when installing in graphical or text mode, it is helpful. If you set up values in the properties file, those values will be the default values in the graphical or text mode dialog.

For instructions on setting up this file, refer to Setting Up Your Installation Properties File.

To install POView in graphical mode, perform the following steps:

1. Ensure that the WebLogic server is running and the Database is accessible.
2. If you are viewing the installer from a Windows client:
   - On the Windows client, start an Xserver program that enables you to emulate the X terminal.
   - On the application server, set the display for the Windows client where you want the Oracle Installer to display as follows:

     ```
     export DISPLAY=<IP address>:0.0
     ```
3. From your application server machine, enter the following command:

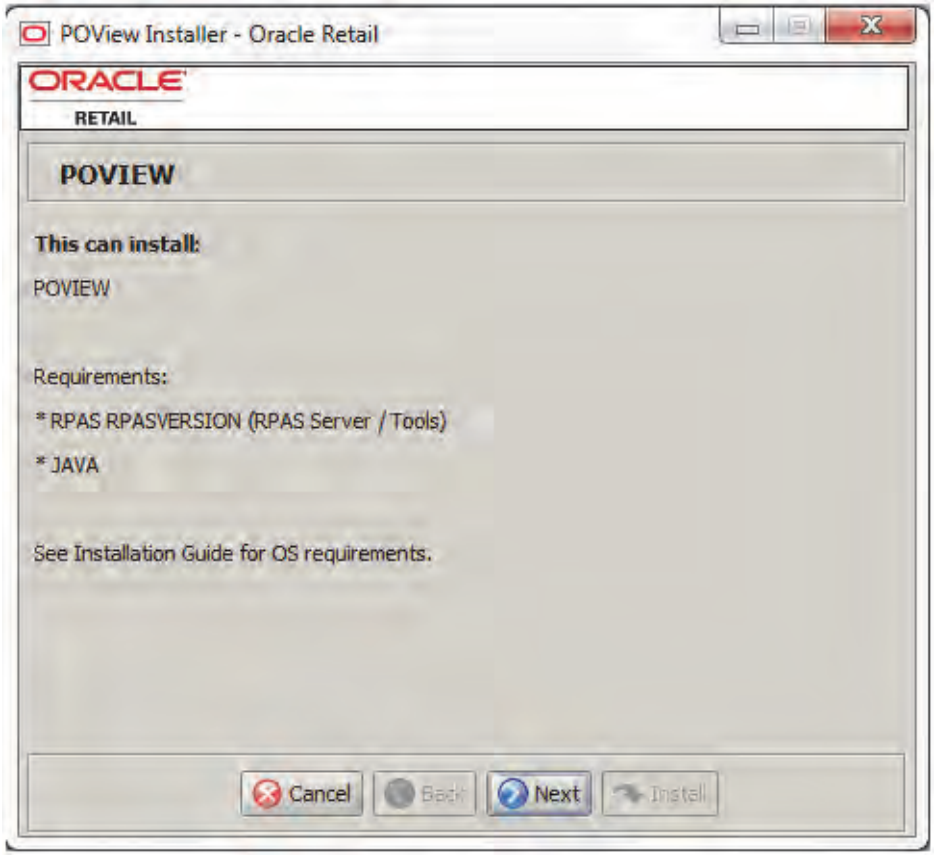
   ```
   ./install.sh
   ```

   When the installer runs, it attempts to validate the version of WebLogic and the associated Oracle Application Development Framework that are used by the WebLogic domain. If you encounter a validation error, fix the problem and try again.

   Note: If you want, you can skip these validations by setting and exporting SKIP_FC_VALIDATION=1 before running the installer.

4. The POView Window opens. Click **Next**.

Figure E–3 POView Window



5. The Retrieve Credentials? Window opens.

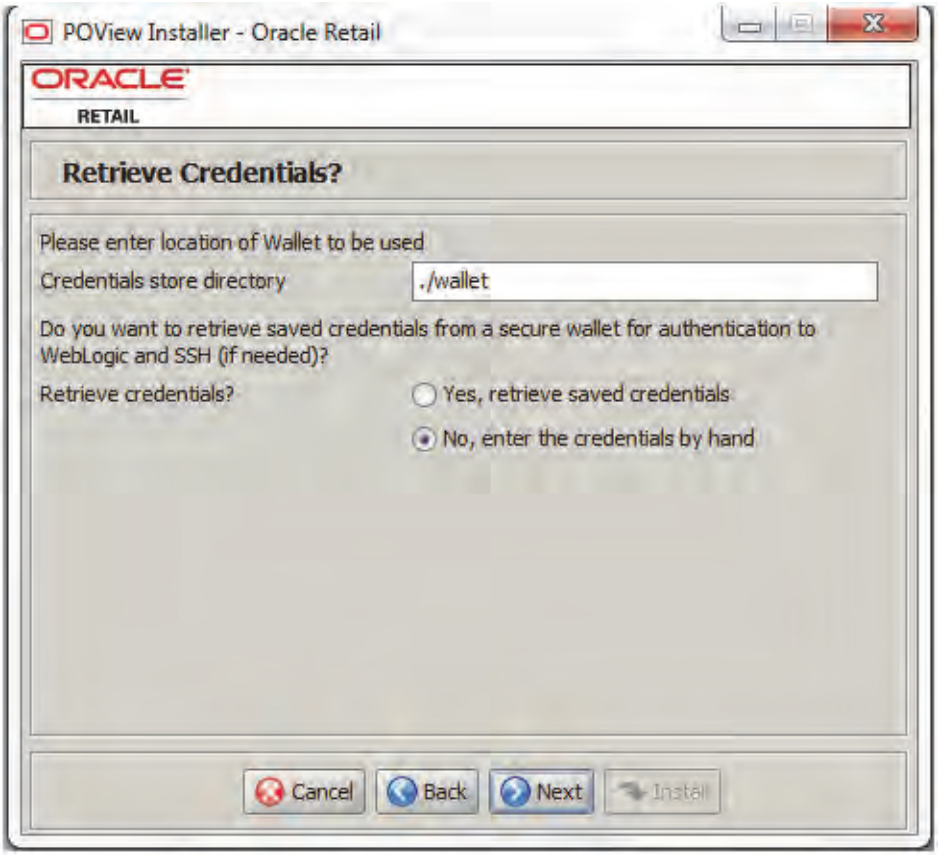
   a. In the Credentials store directory text box, specify the location of the Oracle Wallet you want to use to save or retrieve user credentials.

   b. Select whether to retrieve user credentials from the Oracle Wallet:

Option	Description
Yes, retrieve saved credentials	Indicates that the installer will read the user credentials from an Oracle Wallet
No, enter the credentials by hand	Indicates that you will enter the user credentials in the user interface.

   c. Click **Next**.

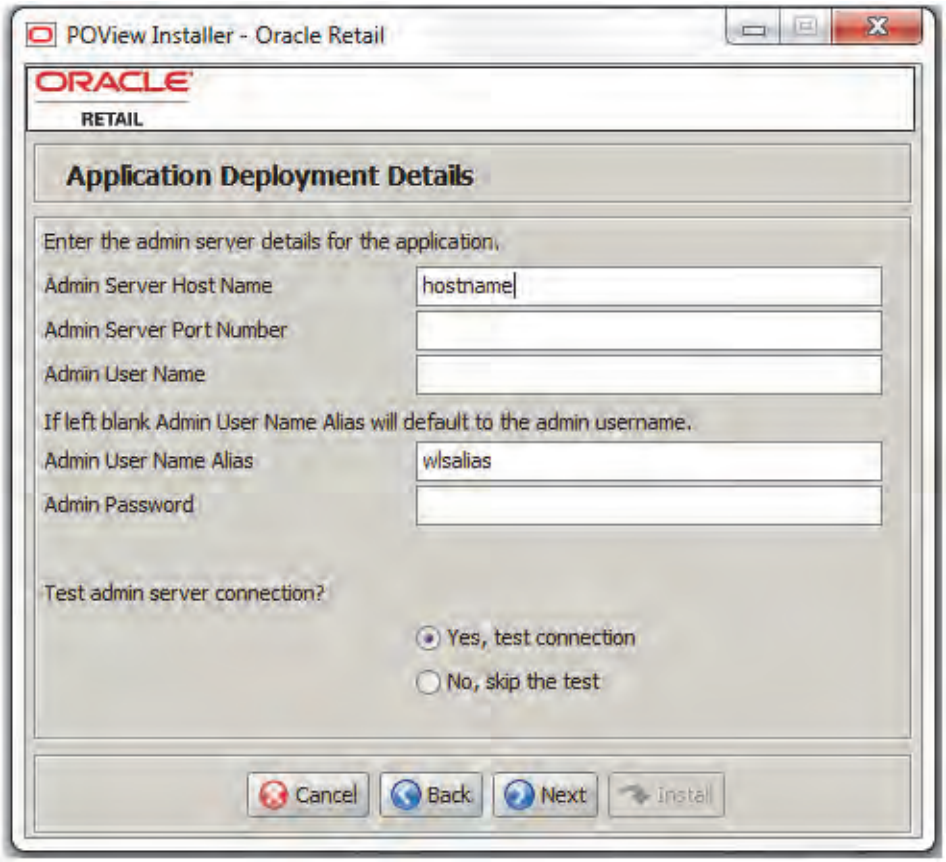
Figure E–4 Retrieve Credentials? Window



6. The WebLogic Admin Server Details Window opens. Enter appropriate information for the following fields and click **Next**:

Field	Description
Admin Server Host Name	Specify the host name of the application server.
Admin Server Port Number	Specify the port number associated with the application server.
Admin User Name	Specify the administrative user name for the application server.
Admin User Name Alias	Specify an alias name for the administrative user. Specifying this name enhances the security for the application. When left blank, the alias name defaults to the administrative user name.
Admin Password	Specify the password associated with administrative user name.
Test admin server connection?	Select **Yes** if you want to test the connection to the admin server. Select **No** if you want to advance to the next window without testing the connection.

Figure E–5 WebLogic Admin Server Details Window

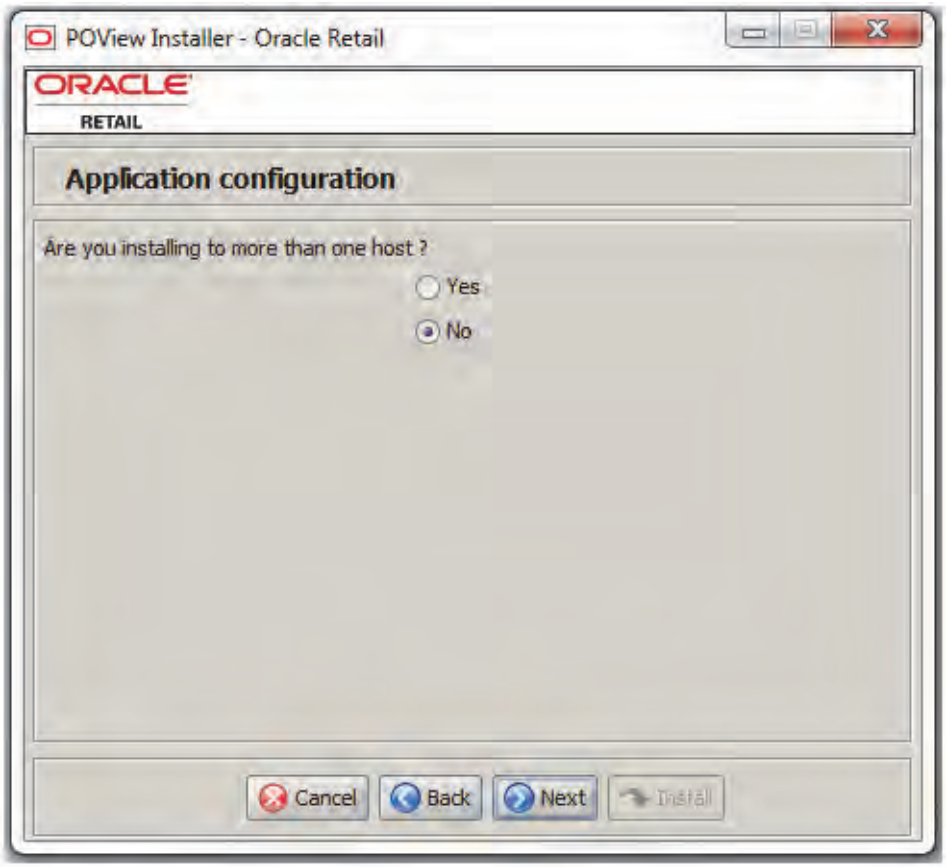


7. The Application Configuration Window opens.

   Specify whether or not you are installing to more than one host:

Option	Description
Yes	This indicates an installation where at least one managed server is running on a remote machine (remote to the admin server machine). Go to Step 8.
No	This indicates an installation where all of the managed servers for deploying the Fusion Client are running on the same machine as the admin server machine, or if you are deploying to the admin server only. Go to Step 9.

Figure E–6 Application Configuration Window



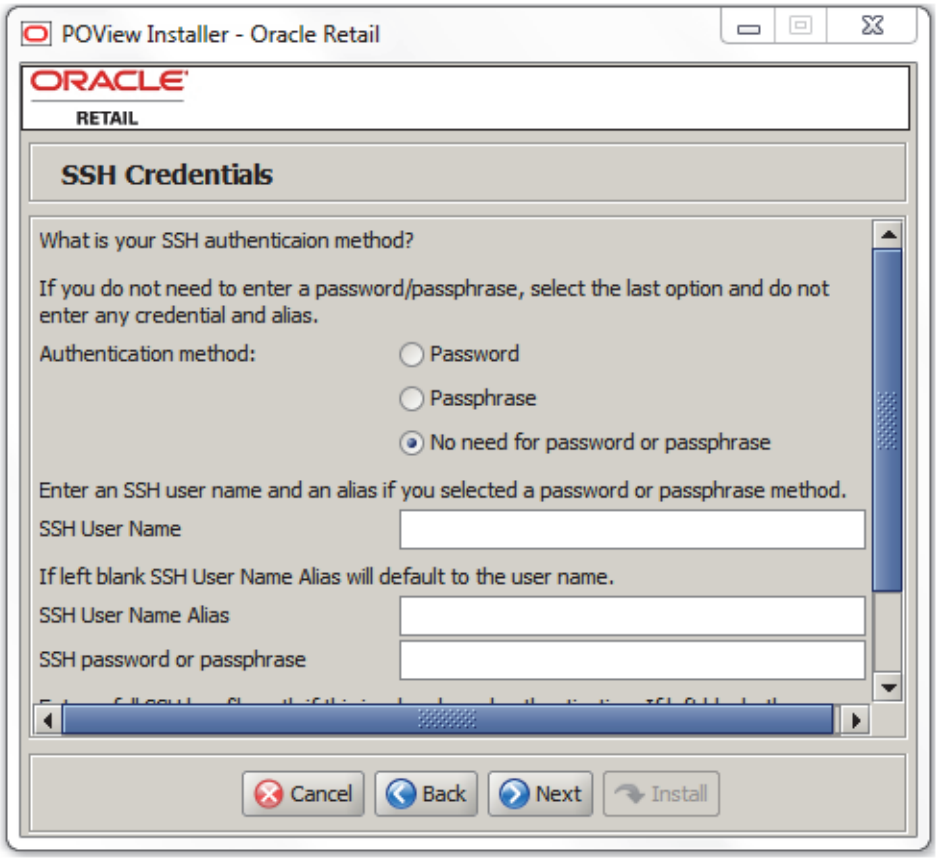
8. If you selected **Yes** on the Application Configuration Window, then the SSH Credentials Window opens, otherwise continue to Step 9

   Enter the relevant information in the following fields and click **Next**:

Field	Description
Authentication method	Select one of the following authentication methods: ▪ **Password** — Use the specified password (associated with the SSH User Name) to connect to the remote hosts for copying the files. ▪ **Passphrase** — Use the specified passphrase (associated with the SSH User Name) along with the SSH Key to connect to the remote hosts. ▪ **No need for password or passphrase** — Default option; connect to the remote hosts without a user name, password, or passphrase.
SSH User Name	Specify the SSH user name to connect to the remote hosts.
SSH User Name Alias	Specify the alias name associated with the SSH user name. Specifying an alias name enhances the security for the application. When left blank, the alias name will default to the administrative user name.
SSH password or passphrase	Based on the authentication method you selected, enter the relevant SSH password or passphrase.

Field	Description
SSH Key File Path	In case you selected the Passphrase option in the Authentication Method field, enter the location of the SSH key file. When left blank, the installer will retrieve the file from ${user.home}/.ssh/id_dsa directory, where user.home is your home directory. To use this default location, ensure that you have the private DSA key stored at this location.

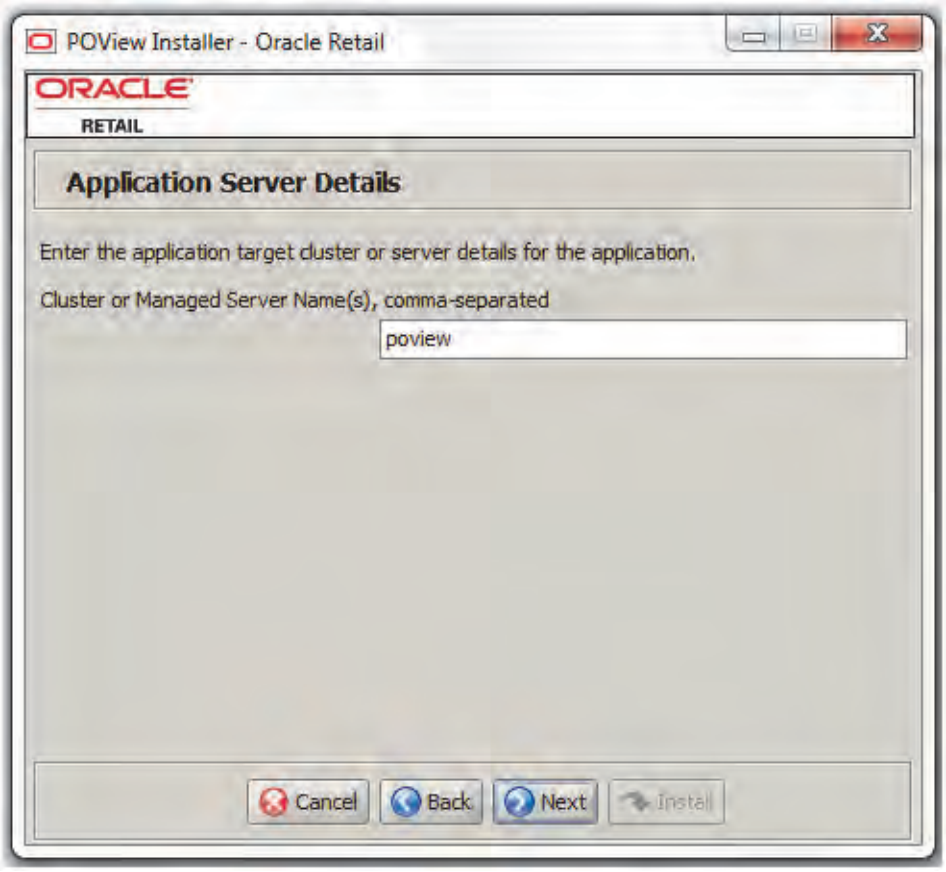
Figure E–7 SSH Credentials Window



9. The Application Server Details Window opens. Enter relevant information for the following fields and click **Next**:

Field	Description
Cluster or Managed Server Name(s), comma-separated	Enter the cluster or managed server names where you want to deploy the fusion client.

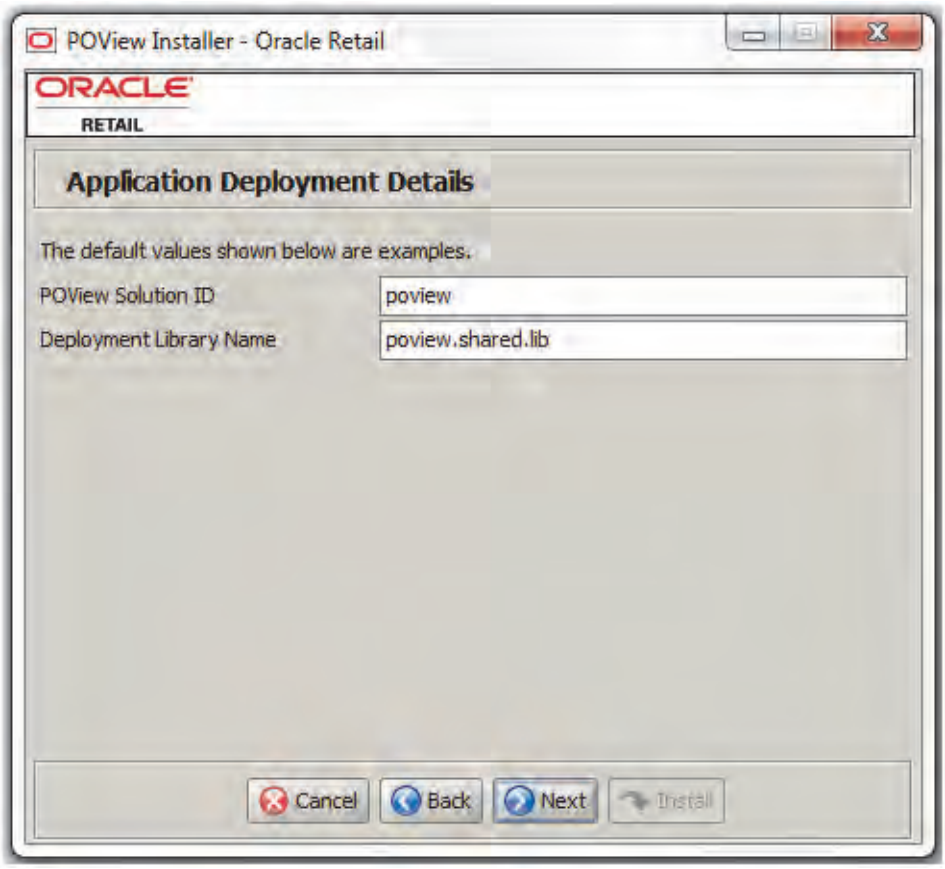
Figure E–8 Application Server Details Window



10. The Application Deployment Details Window opens. Enter relevant information for the following fields and click Next:

Field	Description
POView Solution ID	Enter the solution ID for POView
Deployment Library Name	Enter the deployment library name: `poview.shared.lib`

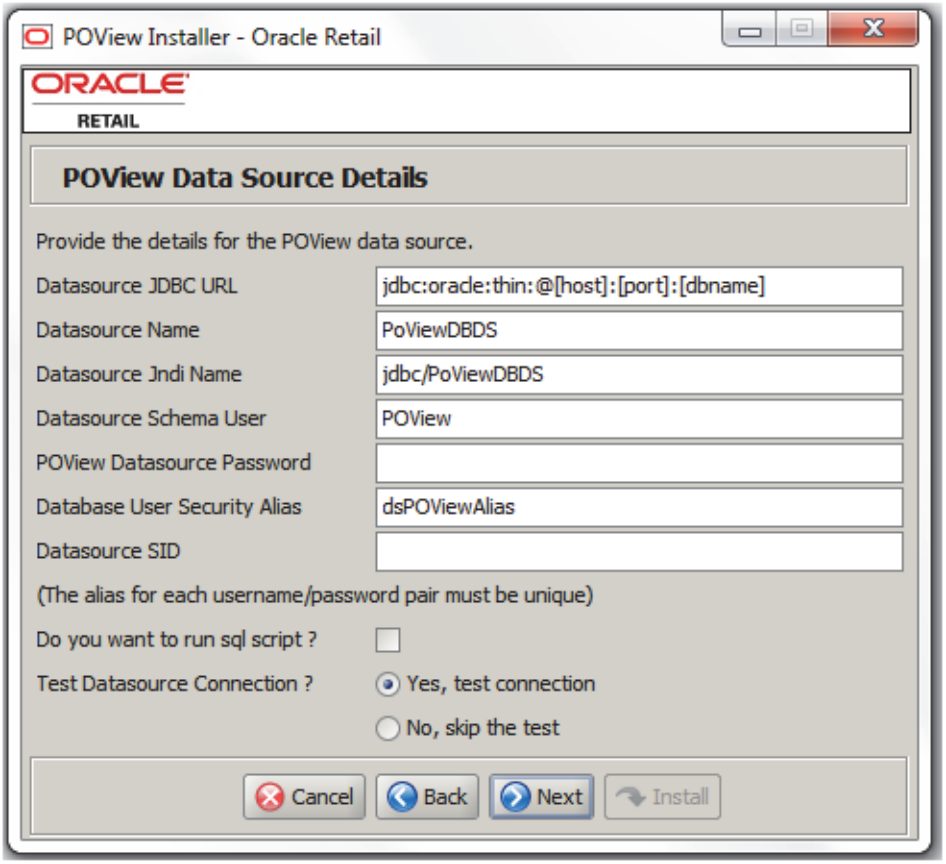
Figure E–9 Application Deployment Details Window



11. The Application Data Source Details Window opens. Enter relevant information for the following fields and click **Next**:

Field	Description
Datasource JDBC URL	Enter the JDBC URL in the format: `jdbc:oracle:thin:@[host]:[port]:[dbname]`
Datasource Name	Enter the name of the Datasource as: `PoViewDBDS`
Datasource Jndi Name	Enter the JNDI name. For example, `jdbc/PoViewDBDS`
Datasource Schema user	Enter the schema user name.
POView Datasource Password	Enter the password for the schema.
Database User Security Alias	Enter the security alias for database user.
Datasource SID	Enter the System ID for the Database.
Do you want to run sql script?	Select the check box to run sql.
Test Datasource Connection	Select to either test the connection or skip the test

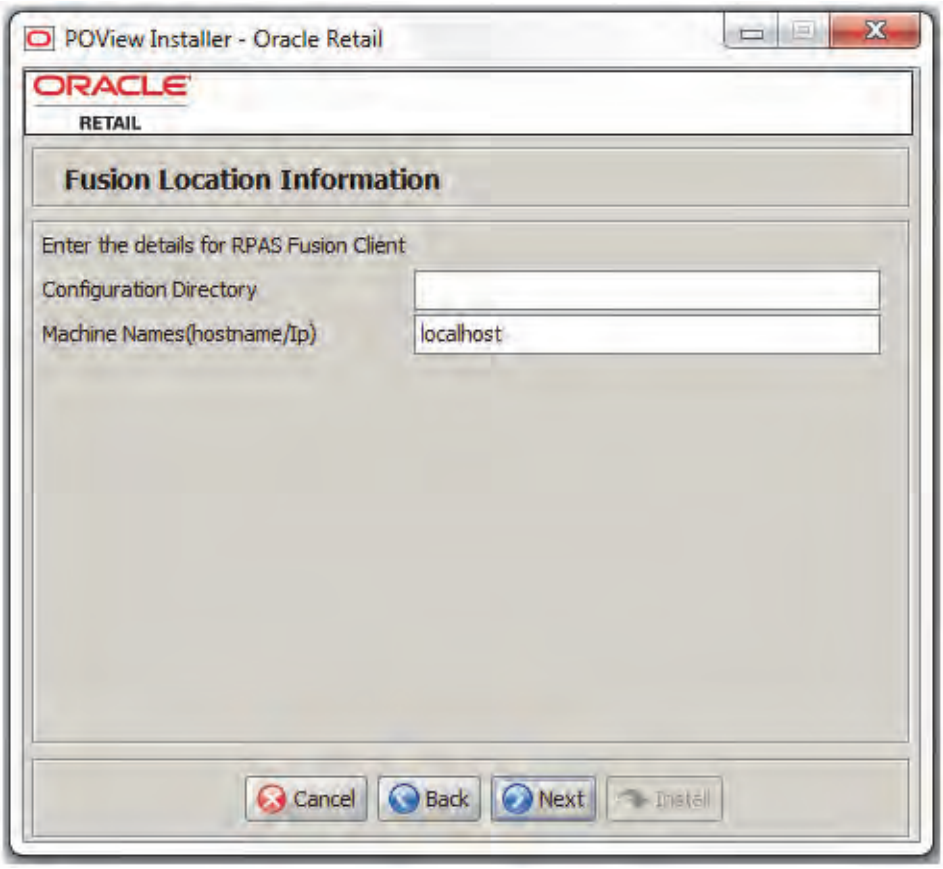
Figure E–10 Application Data Source Details Window



12. The Fusion Location Information Window opens. Enter relevant information for the following fields and click **Next**:

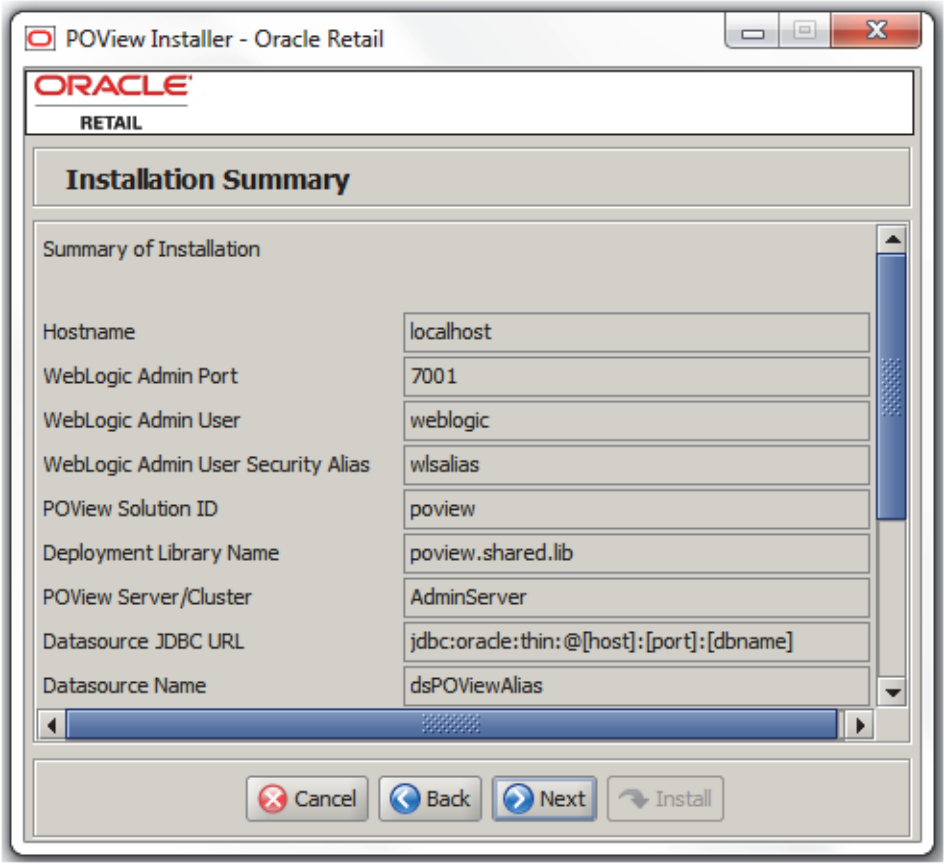
Field	Description
Configuration Directory	Specify the location where the RPAS Fusion Client is installed.
Machine Names (hostname/Ip)	Enter the hostname/IP of the machine where Fusion Client is installed.

Figure E–11 Fusion Location Information Window



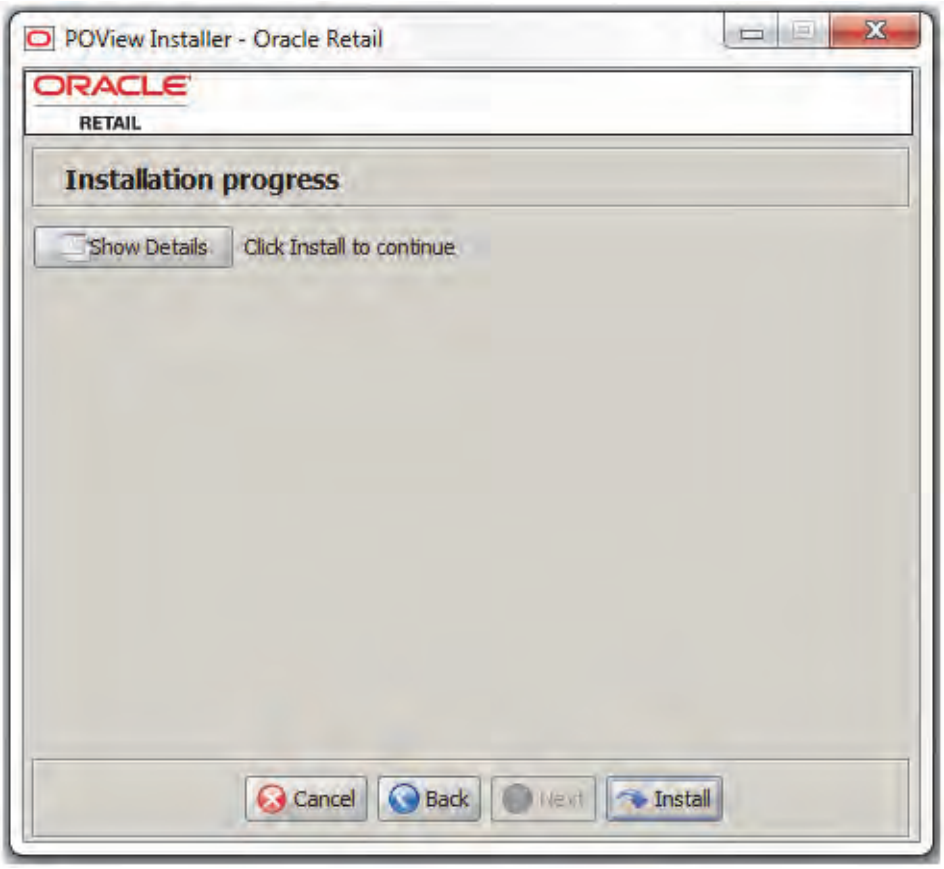
13. The Installation Summary Window opens. Review the installation summary and click **Next**.

Figure E–12 Installation Summary Window



14. The Installation Progress Window opens. To start the installation, click **Install**.

Figure E–13 Installation Progress Window



15. After the installation is complete, click **Exit** to close the Installer.

Post-Installation Tasks

After successfully completing installation, complete these post-installation tasks:

- Fusion Client Install
- Configuring In-Context Module
- Configuring the Manifest File

Fusion Client Install

For details on Fusion Client setup refer to Chapter 5, "Installing the RPAS Fusion Client" and any Oracle Retail product specific Installation Guide for the Fusion Client.

Note: Edit the ant.install.properties file using any text editor and set

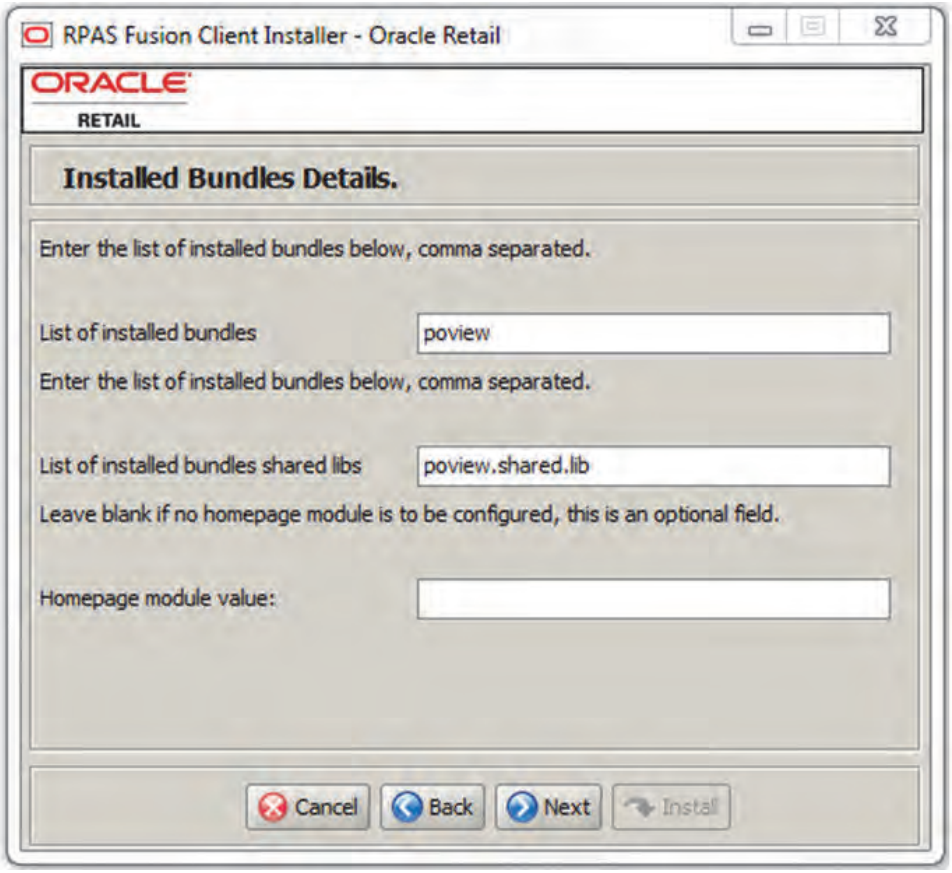
- `input.installed.bundles = poview`
- `input.installed.bundles.shared.libs = poview.shared.lib`

On the RPAS Fusion Client Installer —Installed Bundles Details Window, specify the:

- List of bundles

- List of shared libraries

Figure E–14 Installed Bundles Details Window



Configuring In-Context Module

To launch POView as an in-context module from a worksheet, an XML element is configured in the task flow configuration file, namely Taskflow_MultiSolution.xml. Where the element is inserted determines which worksheet's context menu makes the plug-in available for launch.

The structure of the task flow configuration file can be described briefly as follows: it is a set of activities. Each activity has a set of tasks. A task has a set of steps; a step has a set of worksheets.

The in-context launch entry can be created as a sub-element of a task, step or worksheet. The implication of each insertion point is described in the following table.

Insertion Point	Effect
Task	The plug-in is available on every worksheet of every step under the task.
Step	The plug-in is available in every worksheet under the step.
Worksheet	The plug-in is available on that worksheet.

Update the following items:

- Add the following XML element:

```
<incontext_modules>
   <module name="poview" bundle="poview" label="poview.label"
resource_bundle="oracle.rgbu.ard.util.i18n.SolutionResourceBundle"/>
 </incontext_modules>
```

- Add the following.

  Snippet of POView configured for a Step in Taskflow_MultiSolution.xml.

```
        <step>
          <name>mfprtl.Activity1.Activity1.Task11.Step24</name>

<description>mfprtl.Activity1.Activity1.Task11.Step24.Desc</description>
          <order_num>1</order_num>

<instructions>mfprtl.Activity1.Activity1.Task11.Step24.Instructions</instructio
ns>
          <worksheet>
  <incontext_modules>
      <module name="poview" bundle="poview" label="poview.label"
resource_bundle="oracle.rgbu.ard.util.i18n.SolutionResourceBundle"/>
  </incontext_modules>
            <name>td_initialize1</name>
          </worksheet>
          <custom_menu>91</custom_menu>
        </step>
```

- Add the following property to {RPAS Fusion Client Install Directory}\MultiSolution\resources\MultiSolutionBundle.properties:

```
poview.label=POView
```

- Edit the poview-log4jconfig.xml file under RPAS Fusion Client Install directory and change the value for the property `<param name="file" value="/poview.log"/>` for logging to a different directory-file:

```
<appender name="poviewFileAppender"
class="org.apache.log4j.RollingFileAppender">
  <param name="append" value="true"/>
  <param name="file" value="/poview.log"/>
    <layout class="org.apache.log4j.PatternLayout">
      <param name="ConversionPattern" value="%d{ABSOLUTE} %-5p [%c{1}] %m%n"/>
    </layout>
</appender>
```

Configuring the Manifest File

Each functional module comes with a manifest, using which the bundle configurator can configure the functionality of the bundle within the Fusion Client. The bundle manifest for POView has to have details about Datasource, Selection context and Metrics. The Installer copies the manifest to the functionalmodulebundles\poview directory under the Fusion Client install directory.

Restart the WebLogic server for manifest file changes to take effect.

Datasource

The data source from which POView receives the RMS data is configured through the bundle manifest. On the WebLogic server the same Datasource has to be configured.

The Data source creation on WebLogic and update of manifest with the Datasource details is updated by the POView Installer. The following entry under the bundle manifest is updated by the Installer (it should not be manually changed):

```
<data_sources>
    <data_source name="PoViewDBDS" jndi_name="jdbc/PoViewDBDS"/>
</data_sources>
```

Note: The previous entry is updated by the Installer and this should not be changed manually.

Metrics

The manifest dictates the metrics that are visible in the Item Details panel. The metrics that are available for configuration are:

- Quantity Ordered
- Quantity Received
- Quantity Cancelled
- Quantity Open
- Quantity Ordered Unit Cost
- Quantity Received Retail
- Quantity Cancelled Unit Cost
- Quantity Open Unit Cost

To hide a metric column change the value to N. To show a column, configure the value to Y against the corresponding parameter in the manifest.

```
<param name="QUANTITY_ORDERED_RENDERED" value="Y"/>
  <param name="QUANTITY_RECEIVED_RENDERED" value="Y"/>
  <param name="QUANTITY_CANCELLED_RENDERED" value="Y"/>
  <param name="QUANTITY_OPEN_RENDERED" value="Y"/>
  <param name="QUANTITY_ORDERED_DOLLARS_RENDERED" value="Y"/>
  <param name="QUANTITY_RECEIVED_DOLLARS_RENDERED" value="Y"/>
  <param name="QUANTITY_CANCELLED_DOLLARS_RENDERED" value="Y"/>
  <param name="QUANTITY_OPEN_DOLLARS_RENDERED" value="Y"/>
```

Selection Context

The POView can be configured to accept different selection contexts based on its definition in the manifest. The manifest can be updated to change the selection criteria and its cardinality. On the Product Dimension the available Selection Context's are:

- Departments
- Classes
- Subclasses

On the Location dimension the available Selection Context's are:

- Company
- Chains

- Channels
- Districts
- Regions
- Areas
- Stores

The following entry under the manifest has to be configured.

```
    <position_input>
<dimension rpas_name="clnd" module_name="Calendar" required="false">
<level rpas_name="week" module_name="Week" cardinality="*"/>
<level rpas_name="mnth" module_name="Month" cardinality="*"/>
      </dimension>
      <dimension rpas_name="loc" module_name="Location">
<level rpas_name="chnl" module_name="CHNL" cardinality="*"/>
            <level rpas_name="chn" module_name="CHN" cardinality="*"/>
<level rpas_name="area" module_name="AREA" cardinality="*"/>
            <level rpas_name="rgn" module_name="RGN" cardinality="*"/>
<level rpas_name="dstr" module_name="DSTR" cardinality="*"/>
            <level rpas_name="str" module_name="STR" cardinality="*"/>
<level rpas_name="comp" module_name="COMP" cardinality="1"/>
      </dimension>
      <dimension rpas_name="prod" module_name="Product">
            <level rpas_name="scls" module_name="SCLS" cardinality="*">
<parent_level rpas_name="clss" module_name="CLSS"/>
<parent_level rpas_name="dept" module_name="DEPT"/>
            </level>
            <level rpas_name="clss" module_name="CLSS" cardinality="*">
<parent_level rpas_name="dept" module_name="DEPT"/>
            </level>
<level rpas_name="dept" module_name="DEPT" cardinality="*"/>
       </dimension>
       <date_range/>
    </position_input>
```

Note: The cardinality of Company level in the Location dimension has to be 1 and cannot be configured to anything else. Cardinality for other levels can be 1 or *. Cardinality provides a way to select one or more positions on a certain level. For example, if the Department level is configured with cardinality 1, on the Worksheet only one position for Department level can be selected. However, if Department is configured with cardinality as '*' then one or more positions for Department level can be configured.

Selection Context Use Cases

Use Case 1: Allow Selection at only Department Level on Product Dimension

To specify a selection at the Department level alone on the Product dimension, use the following configuration:

```
<dimension rpas_name="prod" module_name="Product">
  <level rpas_name="dept" module_name="DEPT" cardinality="*" />
</dimension>
```

Use Case 2: Allow Selection for only a Single Department on Product Dimension

To specify a selection for a single Department on the Product dimension, use the following configuration:

```
<dimension rpas_name="prod" module_name="Product">
  <level rpas_name="dept" module_name="DEPT" cardinality="1" />
</dimension>
```

Use Case 3: Allow Selection at Department/Class Level on Product Dimension

To specify both Department and Class on the Product dimension, use the following configuration:

```
<dimension rpas_name="prod" module_name="Product">
  <level rpas_name="clss" module_name="CLSS" cardinality="*">
     <parent_level rpas_name="dept" module_name="DEPT"/>
  </level>
  <level rpas_name="dept" module_name="DEPT" cardinality="*" />
</dimension>
```

Use Case 4: Allow Selection at Channel and Chain Level on Location Dimension

To specify both Channel and Chain on the Location dimension, use the following configuration:

```
<dimension rpas_name="loc" module_name="Location">
  <level rpas_name="chnl" module_name="CHNL" cardinality="*"/>
  <level rpas_name="chn" module_name="CHN" cardinality="*"/>
</dimension>
```

Use Case 5: Allow Selection for a Single Chain on Location Dimension

To specify a selection for a single Chain on the Location dimension, use the following configuration:

```
<dimension rpas_name="loc" module_name="Location">
  <level rpas_name="chn" module_name="CHN" cardinality="1"/>
</dimension>
```

Use Case 6: Hiding the Quantity Ordered and Quantity Received Metrics

```
<param name="QUANTITY_ORDERED_RENDERED" value="N"/>
      <param name="QUANTITY_RECEIVED_RENDERED" value="N"/>
```

F

Appendix: Installation Order

This section provides a guideline for the order in which the Oracle Retail applications should be installed. If a retailer has chosen to use only some of the applications, the order is still valid, less the applications not being installed.

> **Note:** The installation order is not meant to imply integration between products.

Enterprise Installation Order

1. Oracle Retail Merchandising System (RMS), Oracle Retail Trade Management (RTM)
2. Oracle Retail Sales Audit (ReSA)
3. Oracle Retail Extract, Transform, Load (RETL)
4. Oracle Retail Active Retail Intelligence (ARI)
5. Oracle Retail Warehouse Management System (RWMS)
6. Oracle Retail Invoice Matching (ReIM)
7. Oracle Retail Price Management (RPM)

> **Note:** During installation of RPM, you are asked for the RIBforRPM provider URL. Since RIB is installed after RPM, make a note of the URL you enter. If you need to change the RIBforRPM provider URL after you install RIB, you can do so by editing the remote_service_locator_info_ribserver.xml file.

8. Oracle Retail Allocation
9. Oracle Retail Central Office (ORCO)
10. Oracle Retail Returns Management (ORRM)
11. Oracle Retail Back Office (ORBO)

12. Oracle Retail Store Inventory Management (SIM)

    Note: During installation of SIM, you are asked for the RIB provider URL. Since RIB is installed after SIM, make a note of the URL you enter. If you need to change the RIB provider URL after you install RIB, you can do so by editing the remote_service_locator_info_ribserver.xml file.

13. Oracle Retail Predictive Application Server (RPAS)
14. Oracle Retail Demand Forecasting (RDF)
15. Oracle Retail Category Management Planning and Optimization/Macro Space Optimization (CMPO/MSO)
16. Oracle Retail Replenishment Optimization (RO)
17. Oracle Retail Analytic Parameter Calculator Replenishment Optimization (APC-RO)
18. Oracle Retail Regular Price Optimization (RPO)
19. Oracle Retail Merchandise Financial Planning (MFP)
20. Oracle Retail Size Profile Optimization (SPO)
21. Oracle Retail Assortment Planning (AP)
22. Oracle Retail Item Planning (IP)
23. Oracle Retail Item Planning Configured for COE (IP COE)
24. Oracle Retail Advanced Inventory Planning (AIP)
25. Oracle Retail Analytics
26. Oracle Retail Advanced Science Engine (ORASE)
27. Oracle Retail Integration Bus (RIB)
28. Oracle Retail Service Backbone (RSB)
29. Oracle Retail Financial Integration (ORFI)
30. Oracle Retail Point-of-Service (ORPOS)
    - Oracle Retail Mobile Point-of-Service (ORMPOS) (requires ORPOS)
31. Oracle Retail Markdown Optimization (MDO)
32. Oracle Retail Clearance Optimization Engine (COE)
33. Oracle Retail Analytic Parameter Calculator for Markdown Optimization (APC-MDO)
34. Oracle Retail Analytic Parameter Calculator for Regular Price Optimization (APC-RPO)
35. Oracle Retail Macro Space Planning (MSP)

    The Oracle Retail Enterprise suite includes Macro Space Planning. This can be installed independently of and does not affect the installation order of the other applications in the suite. If Macro Space Planning is installed, the installation order for its component parts is:

    - Oracle Retail Macro Space Management (MSM)

- Oracle Retail In-Store Space Collaboration (ISSC) (requires MSM)
- Oracle Retail Mobile In-Store Space Collaboration (requires MSM and ISSC)