Oracle® Hyperion Enterprise Performance Management System

Manual Deployment Guide

RELEASE 11.1.1.4

Updated: November 2011

ORACLE

ENTERPRISE PERFORMANCE MANAGEMENT SYSTEM

EPM System Manual Deployment Guide, 11.1.1.4

Authors: EPM Information Development Team

Contents

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 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.

1 Getting Started with Manual Deployment

In This Chapter

Check the Oracle Documentation Library http://www.oracle.com/technology/documentation/epm.html on Oracle® Technology Network to see whether an updated version of this guide is available.

Audience

This guide includes information that is required to manually deploy the Oracle Hyperion Enterprise Performance Management System. This guide is intended for experienced Web application server administrators who are responsible for installing, deploying, and configuring Web applications.

Skills Required for Manual Deployment

Manual deployment should be performed only by administrators who are highly skilled at the following tasks:

- Preparing Web application server environments
- Installing and configuring Web application servers
- Installing and configuring Web server plug-ins
- Using the Web application server command line interface (CLI) and administration console to perform administrative tasks
- Working with the Web application server architecture
- Deploying J2EE applications on Web application servers
- Configuring Web application server logging

- Troubleshooting Web application server and Web server issues

About This Guide

Conventions Used in This Guide

- **boldface**—Boldface text indicates, within procedural steps, graphical user interface elements that are associated with actions.
- *italic*—Italic text indicates book titles, emphasis.
- *ITALIC/CAPITAL LETTERS*—Italic and capital letter text indicates placeholder variables for which you supply values.
- `monospace`—Monospace text indicates commands, URLs, code examples, text that is displayed on the screen, or text that you enter.

Terminology Used in This Guide

- **WAR** (**W**eb **AR**chive) file—A collection of JavaServer pages (JSPs), servlets, Java classes, and other related files that comprise a Web application
- **EAR** (Enterprise **AR**chive) file—A collection of one or more WAR files, plus deployment descriptors that describe how to deploy the files on a Web application
- **Instance**—A Java container provided by a Web application server; for example, OC4J instance, WebLogic managed server instance, WebSphere application server instance

Location References Used in This Guide

This guide uses the following variables to identify locations.

Variable	Location
Hyperion home directory	The directory into which EPM System products should be installed. By default, it is Windows: `C:\Hyperion`. UNIX: `/home/`*`username`*`/Hyperion`
HYPERION_HOME	A system variable for the Hyperion home directory.
`HYPERION_HOME`	A code variable for the Hyperion home directory used in code examples.
`DEPLOYED_LOCATION`	The directory to where the Web archive was expanded; for example, *`ORACLE_HOME`*`/j2ee`
`LISTEN_PORT`	The TCP port number on which the deployed application accepts requests. For example, your Web server listen port. For a list of default ports, see *Oracle Enterprise Performance Management System Installation Start Here*.

Variable	Location
ORACLE_HOME	The directory where Oracle Application Server is installed. Oracle Application Server Oracle Application Server
INSTANCE	The name of an OC4J instance.
WL_HOME	The directory where Oracle WebLogic Server is installed.
DOMAIN_HOME	The directory where your WebLogic domain is installed.
SERVER_NAME	The name of a WebLogic managed server.
WAS_ROOT	The directory where WebSphere Application Server is installed.
PROFILE_NAME	The name of a WebSphere profile.

Specifying Web Application Server Hostnames and Web Server Connector Plug-in Ports

When EPM System is distributed among two or more machines or when an EPM System Web application is configured to accept requests from a machine other than the machine where it is installed, you must provide the correct server hostnames and listen ports. You need to specify the correct ports for all installation scenarios, even for a single-host installation. Specify the machine name and port as part of the Oracle's Hyperion Enterprise Performance Management System Configurator process.

Note: Use fully qualified hostnames for all entries; for example, hostname.example.com.

Understanding EPM System Web Components

Web Server

Oracle Enterprise Performance Management Workspace, Fusion Edition provides a rich, modular DHTML user interface. Modules run in their own Web application server instances, which affords the flexibility to distribute workload among multiple hosts for better performance, scalability and fail-over.

- For security, Web browsers allow only pages that originate from the same host to interact. All EPM System components that integrate with EPM Workspace must be accessible, therefore, using the same server hostname and port. A Web server is used to fulfill this requirement.
- Using vendor-provided Web server plug-ins, the Web server used by the EPM Workspace acts as a reverse proxy to route client requests. Requests may be handled in part by the Web

server itself, such as requests for static images that exist on the same machine as the Web server, or routed to the appropriate Web application server instance for processing. Requests to stand-alone EPM System components that do not integrate with the EPM Workspace, such as Oracle's Hyperion® Shared Services, can also be routed using the Web server proxy.

- The Web server can serve static content, which reduces the load on the Web application server (EPM Workspace and Oracle's Hyperion® Web Analysis are configured to do this).
- The Web server makes it easier to support security requirements, such as firewalls and SSL.
- The Web server is required to support Single Sign On solutions, such as Oracle Single Sign On (OSSO).

Example That Uses a Web Server

The following example illustrates the role played by the Web server:

- EPM Workspace—Deployed to the Oracle Application Server OC4J instance on `machine1.example.com` and configured to accept AJP requests
- Oracle Hyperion Financial Reporting, Fusion Edition Web component—Deployed to the Oracle Application Server OC4J instance on `machine2.example.com`, and configured to accept AJP requests
- Web server (Oracle HTTP Server)—Installed on `machine3.example.com`, and configured to accept incoming HTTP requests on port 19000

  In this configuration, the mod_oc4j Web server plug-in is configured for Oracle HTTP Server (OHS) on `machine3.example.com`. It will route dynamic content requests to the `/workspace` context to OC4J on `machine1.example.com`, and will route all requests to the `/hr` context to `machine2.example.com`. Static content requests, such as images, to `/workspace` is served directly by OHS from a directory local to `machine3.example.com`.

Users of this system will access EPM Workspace by using the URL:

```
http://machine3.example.com:19000/workspace/
```

EPM Workspace processes a request for a Financial Reporting document by using the following URL:

```
http://machine3.example.com:19000/hr/
```

Example That Does Not Use a Web Server

The following example illustrates what happens when a Web server is not used:

- EPM Workspace—Deployed to the Oracle Application Server OC4J instance on `machine1.example.com` and configured to accept HTTP requests on port 45000, bypassing OHS
- Financial Reporting—Deployed to the Oracle Application Server OC4J instance on `machine2.example.com` and configured to accept HTTP requests on port 8200, bypassing OHS

In this configuration, EPM Workspace is accessed using:

`http://machine1.example.com:45000/workspace/`

EPM Workspace constructs URIs by using:

`http://machine1.example.com:45000/hr/`

and retrievals of Financial Reporting documents fail.

Web Application Server

The Web-enabled components of the EPM System are mostly composed of JavaServer pages (JSPs), Servlets, HTML text, and other related content. However, some exceptions exist. For example, when Windows and IIS are used, Oracle Hyperion Financial Management, Fusion Edition implements ASP technology.

Note: This guide focuses only on the components powered by a Java application server.

Dependent Java Libraries

All EPM System Web applications rely on common Java libraries to provide basic functionality, such as user authentication and authorization, logging, database connectivity, and data source connectivity. In some cases, dependent Java libraries are installed outside the application server under the Hyperion home directory. The Web applications find and load the external libraries by including them in the CLASSPATH of the Web application server instance. Other libraries are packaged with the application EAR or WAR file and deployed directly on the application server.

Various factors (such as whether a library is needed for JSP compilation, application runtime, or both JSP compilation and application runtime), determine where dependent Java libraries are installed. EPM System Web applications are not "self-contained" in their respective EAR or WAR files. The Oracle Hyperion Enterprise Performance Management System Installer, Fusion Edition must be run on each Web application server machine that hosts an EPM System Web application. For a Web application cluster, the EPM System Installer must be run on each physical node of the cluster, and the Hyperion home directory should be installed to the same file system path on each machine.

Dependent Native Libraries

Understanding Native Libraries

Several EPM System Web applications use the Java Native Interface (JNI) framework to call native applications and libraries that are written in other languages, such as C and C++. The Web applications find and load the native components by including the components in the library search path (`java.library.path`) that is used by the Web application server instance to which they are deployed.

Most often, to include the components in the `java.library.path`, the library search path variable that is specific to the operating system is set in a start script, service definition, or application server configuration file. It is not recommended that library paths be added as global shell or operating system variables, as applications deployed to the same machine may require different versions of the same library.

Library Search Path Variables

This table lists the operating system and the corresponding library search path variable.

OS	Path Variable
Windows	PATH
Linux, Solaris, HP-UX (IA64)	LD_LIBRARY_PATH
AIX	LIBPATH
HP-UX	SHLIB_PATH

Many of the native libraries used by EPM System products, some of them belonging to third parties, have only 32-bit versions available. For this reason, unless otherwise noted, EPM System products support only 32-bit versions of the application servers.

Note: Oracle Essbase Administration Services, Oracle Hyperion Provider Services, and Oracle Hyperion Planning, Fusion Edition support 64-bit Web application servers with manual deployment.

Note: It is valid and supported to install and use 32-bit versions of supported application servers on 64-bit operating systems.

Manual Deployment Process Overview

Manually deploying an EPM System Web application component to an application server requires these steps:

1. Complete prerequisites
2. Customize the application server instance, as applicable:
   - Configure listen ports and protocols
   - Configure Java heap
   - Configure Java arguments
   - Configure external CLASSPATH
   - Configure the library search path variable. See "Library Search Path Variables" on page 18.

- Configure environment variables

3. Deploy the WAR or EAR file
4. Configure the Web server and Web server plug-in

2 Setting Up Your System for Manual Deployment: Required Steps

In This Chapter

Installing and Configuring the EPM System for Manual Deployment

For information on Installing and Configuring the EPM System, see the *Oracle Enterprise Performance Management System Installation and Configuration Guide*.

The following tasks must be performed prior to deploying EPM System components:

1. Run the EPM System Installer and EPM System Configurator for Shared Services components. During configuration, for deployment type, select **Manual**.
2. Deploy and start Shared Services.
3. Run the Oracle Hyperion Enterprise Performance Management System Installer, Fusion Edition on each application server machine, including on each physical node of an application server cluster. When it is planned that an application will be horizontally clustered, install to the same file system location on each machine. Using the same file system path on each physical machine in a cluster is important so that CLASSPATH and library search path settings can be set once for the entire cluster, rather than be set and customized for each node in the cluster.

   Note: The EPM System components and application server must not be installed to a file system location containing spaces.
4. Run EPM System Configurator on each application server machine. Select the Manual option in the Web application deployment panel; when prompted, specify the correct hostname and port information, and, as needed, use the Advanced Set up feature to configure access using a logical address.

   Note: Planning logical address is defined using the "Manage Planning Clusters" task in the EPM System Configurator.

Consider this example:

- Shared Services installed on `machine1.example.com`, and deployed to an application server instance that listens on HTTP port 28080
- Web server installed on `machine2.example.com`, and configured to accept requests on port 80
- Web server plug-in configured on `machine2.example.com` to route requests to Shared Services on `machine1.example.com`

In this scenario, you would use the Advanced Setup feature to define a logical address for Shared Services of `machine2.example.com:80`. For EPM System Web components that integrate with EPM Workspace, you must also select the EPM Workspace **Configure Web Server** task. You must re-run this task each time a new component is installed and configured.

5. Create an application server instance (OC4J instance, WebLogic managed server, or WebSphere application server) for the application being deployed. Deploying multiple Web applications to one instance may yield unsuccessful results.

   If different operating system (OS) accounts are used to install and run EPM System and your Java application server, the Web application server OS account must be granted:

   - Read access to Hyperion home directory, and to all subdirectories and files therein
   - Write access to *`HYPERION_HOME`*`/logs`

Configuring Memory Settings

Newly created application server instances use default memory settings, which are often too small to accommodate EPM System requirements. Optimal settings can be determined only by close monitoring of application server performance under peak load. Use the following settings as a starting point, and then, after careful testing, adjust as needed.

This table lists the memory setting, the suggested value, and the Java argument.

Setting	Suggested Value	Java Argument
Minimum heap	256 MB	`-Xms256m`
Maximum heap	512 MB	`-Xmx512m`

For application servers that use Sun's Java Virtual Machine (JVM), the size of the permanent generation heap must be increased. These settings should not be applied if using an application server powered by BEA JRockit JVM, or IBM's JVM.

This table lists the memory setting, the suggested value, and the Java argument for Sun's JVM.

Setting	Suggested Value	Java Argument
Minimum permanent generation heap	64 MB	`-XX:PermSize=64m`

Setting	Suggested Value	Java Argument
Maximum permanent generation heap	128 MB	`-XX:MaxPermSize=128m`

Creating Temporary File Storage

For security purposes, change the default location for the temporary files created by EPM System applications to a known, secure location. This change can be accomplished by passing the following Java argument, `-Djava.io.tmpdir=`*`TMPDIR`*.

This table lists the settings for temporary files.

Setting	Suggested Value	Java Argument
Temporary file location	*`HYPERION_HOME`*`/temp`	`-Djava.io.tmpdir=`*`HYPERION_HOME`*`/temp`

Note: You must manually create *`HYPERION_HOME`*`/temp`.

3 Deploying EPM System into Oracle Application Server

In This Chapter

OC4J group administrative settings such as environment variables are shared by all group members. The OC4J instance you create for the Web application being deployed should be assigned to a group created specifically for that application. (When an application is deployed to more than one OC4J instance, those instances can be assigned to the same group.)

For example, when deploying Shared Services and EPM Workspace, the OC4J instance created for Shared Services should be assigned to a different group than the instance created for EPM Workspace. Refer to the *Oracle Process Manager and Notification Server Administrator's Guide* for more information about groups.

Deploying Shared Services

- "Configuring OC4J" on page 26
- "Deploying interop.war" on page 26
- "Verifying Deployment" on page 27

Configuring OC4J

➤ To configure OC4J:

1 **Start OpenLDAP using the Windows service, or by running**

`HYPERION_HOME/products/Foundation/openLDAP/startService.bat|startOpenLDAP.sh`

2 **Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".**

3 **Add the following Java arguments**

```
-DHYPERION_HOME=HYPERION_HOME
-Dhyperion.home=HYPERION_HOME
```

For Solaris only:

```
-Djava.nio.channels.spi.SelectorProvider=sun.nio.ch.PollSelectorProvider
```

4 **Add the following:**

a. Add an OC4J environment variable HYPERION_HOME.

b. If necessary, add a library search path variable. See "Library Search Path Variables" on page 18.

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- Select the **Append** option
- If SAP is defined in Shared Services as a user directory, add

  `HYPERION_HOME/common/SAP/bin`

Windows example:

```
HYPERION_HOME  C:\Hyperion
PATH   \Hyperion\common\SAP\bin
```

UNIX and Linux example:

```
HYPERION_HOME  /opt/hyperion
LD_LIBRARY_PATH  /opt/hyperion/common/SAP/bin
```

5 **Restart the OC4J instance.**

Deploying interop.war

➤ To deploy `interop.war`:

1 **Deploy the following Web archive to the OC4J instance that you created.**

`HYPERION_HOME/deployments/Oracle10g/SharedServices9/config/interop.war`

2 **During deployment provide:**

- **Application Name**; for example, `SharedServices`

- Do not import the `apache.commons.logging` shared library

3 **Restart the OC4J instance.**

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open:**

`http://`*`HOSTNAME.EXAMPLE.COM:LISTEN_PORT`*`/interop/`

2 **Review the output for the OC4J instance in** *`ORACLE_HOME`*`/opmn/logs`.

3 **Review the product logs in** ***HYPERION_HOME***`/logs/SharedServices9`.

Deploying Administration Services

Configuring OC4J

➤ To configure OC4J:

1 **Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".**

2 **Add the following Java arguments to the Server Start configuration for the managed server:**

```
-Dhyperion.home=HYPERION_HOME
-DHYPERION_HOME=/HYPERION_HOME
-DESS_ES_HOME=/HYPERION_HOME/products/Essbase/eas/server
-DEAS_HOME=HYPERION_HOME/products/Essbase/eas
-DEAS_LOG_LEVEL=5000
-DEAS_LOG_LOCATION=HYPERION_HOME/logs/eas/easserver.log
-DEAS_SERVER_VERSION=11.1.1.4
```

For Solaris only:

```
-Djava.nio.channels.spi.SelectorProvider=sun.nio.ch.PollSelectorProvider
```

3 **Add the following:**

a. Add OC4J environment variables:

```
HYPERION_HOME
EAS_HOME
ESSLANG
ARBORPATH
```

b. Add a library search path variable. See "Library Search Path Variables" on page 18.

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- Select the **Append** option
- If SAP is defined in Shared Services as a user directory, add

  HYPERION_HOME/common/SAP/bin

Windows example:

```
HYPERION_HOME  C:\Hyperion
EAS_HOME       C:\Hyperion\products\Essbase\eas
ESSLANG        English_UnitedStates.Latin1@Binary
ARBORPATH      C:\Hyperion\products\Essbase\eas\server
ESSBASEPATH    C:\Hyperion\products\Essbase\eas\server

PATH  C:\Hyperion\products\Essbase\eas\server\bin;C:\hyperion\common\SAP\bin;C:
\Hyperion\common\CSS\9.5.0.0\bin
```

UNIX and Linux example:

```
HYPERION_HOME  /opt/hyperion
EAS_HOME       /opt/hyperion/products/Essbase/eas
ESSLANG        English_UnitedStates.Latin1@Binary
ARBORPATH      /opt/hyperion/products/Essbase/eas/server
ESSBASEPATH    /opt/hyperion/products/Essbase/eas/server

LD_LIBRARY_PATH  /opt/hyperion/products/Essbase/eas/server/bin:opt/hyperion/common/
SAP/bin
```

4 **Restart the OC4J instance.**

Deploying eas.ear

➤ To deploy `eas.ear`:

1 **Deploy the following enterprise archive to the OC4J instance that you created:**

HYPERION_HOME/products/Essbase/eas/server/AppServer/InstallableApps/common/eas.ear

2 **During deployment, provide:**

- **Application Name**; for example, `EAS`
- Do not import the `apache.commons.logging` shared library

3 **Restart the OC4J instance.**

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open**

http://*HOSTNAME.EXAMPLE.COM:LISTEN_PORT*/easconsole/console.html

2 Log on to Administration Services using the Java Web Start console.

3 Review the output for the OC4J instance in *ORACLE_HOME*/opmn/logs.

4 Review the product logs in *HYPERION_HOME*/logs/eas.

Deploying Provider Services

Configuring OC4J

➤ To configure OC4J:

1 Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".

2 Add the Java arguments:

```
-DESS_ES_HOME=HYPERION_HOME/products/Essbase/aps
```

For Solaris only:

```
-Djava.nio.channels.spi.SelectorProvider=sun.nio.ch.PollSelectorProvider
```

3 Add the following:

a. Add the OC4J environment variable: HYPERION_HOME

b. Add a library search path variable. See "Library Search Path Variables" on page 18.

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- Select the **Append** option
- If SAP is defined in Shared Services as a user directory, add

  HYPERION_HOME/common/SAP/bin

Windows example:

```
HYPERION_HOME   C:\Hyperion
PATH   C:\Hyperion\products\Essbase\aps\bin;c:\Hyperion\common\SAP\bin;C:\Hyperion
\common\CSS\9.5.0.0\bin
```

UNIX and Linux example:

```
HYPERION_HOME   /opt/hyperion
LD_LIBRARY_PATH   /opt/hyperion/products/Essbase/aps/bin:/opt/common/CSS/9.5.0.0/bin
```

4 Restart the OC4J instance.

Deploying aps.war

➤ To deploy `aps.war`:

1 **Deploy the following Web archive to the OC4J instance that you created:**

HYPERION_HOME`/products/Essbase/aps/AppServer/InstallableApps/common/aps.war`

2 **During deployment, provide:**

- **Application Name**; for example, `APS`
- Do not import the `apache.commons.logging` shared library

3 **Restart the OC4J instance.**

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open:**

`http://`*HOSTNAME.EXAMPLE.COM:LISTEN_PORT*`/aps/APS`

2 **Review the output for the OC4J instance in** *ORACLE_HOME*`/opmn/logs`.

3 **Review the product logs in** *HYPERION_HOME*`/logs/aps`.

Deploying EPM Workspace

Configuring OC4J

➤ To configure OC4J:

1 **Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".**

2 **Add the Java argument:**

```
-Dhyperion.home=HYPERION_HOME
-DHYPERION_HOME=HYPERION_HOME
```

For Solaris only:

```
-Djava.nio.channels.spi.SelectorProvider=sun.nio.ch.PollSelectorProvider
```

3 **Add the following:**

a. Add an OC4J environment variable HYPERION_HOME.

b. If necessary, add a library search path variable. See "Library Search Path Variables" on page 18.

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- Select the **Append** option
- If SAP is defined in Shared Services as a user directory, add

  HYPERION_HOME/common/SAP/bin

Windows example:

```
HYPERION_HOME   C:\Hyperion
PATH   C:\Hyperion\common\SAP\bin;C:\Hyperion\common\CSS\9.5.0.0\bin
```

UNIX and Linux example:

```
HYPERION_HOME   /opt/hyperion
LD_LIBRARY_PATH   /opt/hyperion/common/SAP/bin
```

4 **Restart the OC4J instance.**

Deploying workspace.war

➤ To deploy `workspace.war`:

1 **Deploy the following Web archive to the OC4J instance that you created:**

HYPERION_HOME/products/Foundation/workspace/InstallableApps/workspace.war

2 **During deployment, provide:**

- **Application Name**; for example, `Workspace`
- Do not import the `apache.commons.logging` shared library
- If SAP is defined in Shared Services as a user directory, add the following to the CLASSPATH:

  HYPERION_HOME/common/SAP/lib

3 **Restart the OC4J instance.**

Optional: Deploying biplus_webservices.war

➤ To deploy `biplus_webservices.war`:

1 **Deploy the following Web archive to the OC4J instance that you created:**

HYPERION_HOME/products/Foundation/workspace/InstallableApps/biplus_webservices.war

2 **During deployment, provide:**

- **Application Name**; for example, `BIPlusWebServices`
- Do not import the `apache.commons.logging` shared library

3 **Restart the OC4J instance.**

Configuring Web Server Routing

This section describes the steps to configure Web Server Routing for EPM Workspace.

Other products that need manual steps for Web server routing are:

- Web Analysis: "Configuring Web Server Routing" on page 44
- "Configuring BI Publisher with Apache Web Server or Oracle HTTP Server" on page 53
- "Configuring Financial Management With a Web Server" on page 52
- "Configuring Oracle BI EE with Apache Web Server or Oracle HTTP Server" on page 53

To improve performance, and to reduce load on the application server, static content and online help are served by the Web server.

Note: If your Web server is not on the same machine where EPM Workspace was installed, you must manually copy the following folder:*HYPERION_HOME*`/products/Foundation/workspace/AppServer/InstalledApps/workspace_static`.

Oracle HTTP Server or Apache

Note: In Apache 2.2 and later, the default access state for the Web server root directory is set to `Deny`. This setting will prevent static content for EPM Workspace and Web Analysis from displaying. Refer to the Apache Module mod_authz host documentation for information on changing the default access state.

➤ To configure Oracle HTTP Server and Apache routing:

1 **Using a text editor, add the following directives to your Web server configuration file (`httpd.conf`), replacing *HYPERION_HOME* as needed:**

```
# Change modules/mod_expires.so as needed; for example, libexec/mod_expires on UNIX
<IfModule !expires_module>
  LoadModule expires_module modules/ApacheModuleExpires.dll
</IfModule>
# Images are unlikely to change, so force the browser to cache them. 3 months is the
default.
<LocationMatch  /workspace/(themes|thirdparty|img|images|css|media|docs)/.*/.(gif|
jpeg|jpg|png)$ >
 ExpiresDefault "now"
 ExpiresActive on
 ExpiresByType image/gif "now plus 3 month"
 ExpiresByType image/jpg "now plus 3 months"
```

```
 ExpiresByType image/jpeg "now plus 3 months"
 ExpiresByType image/png "now plus 3 months"
</LocationMatch>

Alias /wsmedia "HYPERION_HOME/products/Foundation/workspace/AppServer/InstalledApps/
workspace_static/wsmedia"
Alias /InsightInstaller "HYPERION_HOME/products/Foundation/workspace/AppServer/
InstalledApps/workspace_static/zeroadmin/component/Insight"
```

Tip: If browser clients will use short hostnames or hostname aliases to access EPM Workspace, consider changing the value of `UseCanonicalName` to Off. For more information on `UseCanonicalName`, see http://httpd.apache.org/docs/2.0/mod/core.html#usecanonicalname.

2 **If using OHS based on Apache 1.3, or Apache 1.3:**

a. Add the following directive to your Web server configuration file (`httpd.conf`), replacing *HYPERION_HOME* as needed:

```
Alias /workspace "HYPERION_HOME/products/Foundation/workspace/AppServer/
InstalledApps/workspace_static"
```

b. Append `ohs-routing="false"` to `default-web-site.xml` to the element for EPM Workspace. for example,

i. Open in a text editor:

```
ORACLE_HOME/j2ee/Workspace/config/default-web-site.xml
```

ii. Change:

```
<web-app application="Workspace" name="workspace" load-on-startup="true"
root="/workspace" />
```

to:

```
<web-app application="Workspace" name="workspace" load-on-startup="true"
root="/workspace" ohs-routing="false" />
```

3 **If using OHS2 based on Apache 2.0, or Apache 2.x, add the following directive to your Web server configuration file (httpd.conf), replacing *HYPERION_HOME* as needed:**

```
AliasMatch /workspace/(resources|css|docs|images|img|SmartView|themes|thirdparty|
wsmedia|zeroadmin|CrystalBall)/(.*) HYPERION_HOME/products/Foundation/workspace/
AppServer/InstalledApps/workspace_static/$1/$2
```

4 **Using a text editor, open `mod_oc4j.conf` on the machine where OHS is installed.**

5 **Add the following Oc4jMount directives, which instructs OHS to route requests made to the specified contexts to OC4J:**

```
Oc4jMount /workspace/administration/* DESTINATION
Oc4jMount /workspace/ DESTINATION
Oc4jMount /workspace/browse/* DESTINATION
Oc4jMount /workspace/cdsrpc DESTINATION
Oc4jMount /workspace/conf/* DESTINATION
Oc4jMount /workspace/dataaccess/* DESTINATION
Oc4jMount /workspace/ihtml/* DESTINATION
Oc4jMount /workspace/jobmanager/* DESTINATION
Oc4jMount /workspace/logon DESTINATION
Oc4jMount /workspace/logon/* DESTINATION
```

```
Oc4jMount /workspace/modules/* DESTINATION
Oc4jMount /workspace/media/* DESTINATION
Oc4jMount /workspace/personalpages/* DESTINATION
Oc4jMount /workspace/portletservlet/* DESTINATION
Oc4jMount /workspace/prefs DESTINATION
Oc4jMount /workspace/search DESTINATION
Oc4jMount /workspace/servlet/* DESTINATION
Oc4jMount /workspace/viewmanager/* DESTINATION
Oc4jMount /workspace/wsrp4j/* DESTINATION
Oc4jMount /workspace/BPMContext DESTINATION
Oc4jMount /workspace/DynamicHelp DESTINATION
Oc4jMount /workspace/PDFView DESTINATION
Oc4jMount /workspace/ResourceProxy DESTINATION
Oc4jMount /workspace/*.jsp DESTINATION
Oc4jMount /workspace/*.jsv DESTINATION
Oc4jMount /workspace/*.jsw DESTINATION
Oc4jMount /biplus_webservices DESTINATION
```

6 **Replace *DESTINATION* with one of the following:**

- ajp13_dest
- cluster_dest (this is the default destination type)
- instance_dest

Example:

```
Oc4jMount /workspace/ cluster://Workspace
```

For information about configuring `mod_oc4j`, see the *Oracle HTTP Server Administrator's Guide*.

7 **Restart the OC4J instance and HTTP Server.**

Customizing EPM Workspace Services Configuration Scripts

EPM Workspace Services include scripts that can be launched interactively to configure various part of the system. When the Manual option is selected during EPM Workspace deployment, the *DEPLOYMENT_HOME* variable declarations must be manually defined in:

HYPERION_HOME`/products/Foundation/workspace/bin/settrustedpass.bat|sh`

➤ To declare the variable declarations:

1 **In a text editor, open:**

HYPERION_HOME`/products/Foundation/workspace/bin/settrustedpass.bat|sh`

2 **Replace occurrences of the *DEPLOYMENT_HOME*=$J(trustedPass.deploymentHome) with:**

```
set DEPLOYMENT_HOME=DEPLOYED_LOCATION
```

where *DEPLOYMENT_HOME* is the file system path to the deployed EPM Workspace Web application.

Example:

```
set DEPLOYMENT_HOME=ORACLE_HOME/j2ee/Workspace/applications/Workspace/workspace
```

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open:**

`http://HOSTNAME.EXAMPLE.COM:LISTEN_PORT/workspace/`

2 **Review the output for the OC4J instance in** *ORACLE_HOME*`/opmn/logs`.

3 **Review the product logs in:**

HYPERION_HOME`/logs/workspace`

Deploying Performance Management Architect

Deploying Performance Management Architect Web

Configuring OC4J

➤ To configure OC4J:

1 **Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".**

2 **Add the Java arguments:**

-Dhyperion`.home=`*HYPERION_HOME*

-Djavax`.xml.stream.XMLOutputFactory=com.ctc.wstx.stax.WstxOutputFactory`

-Djavax`.xml.stream.XMLInputFactory=com.ctc.wstx.stax.WstxInputFactory`

For Solaris only:

-Djava`.nio.channels.spi.SelectorProvider=sun.nio.ch.PollSelectorProvider`

3 **If necessary, add a library search path variable. See "Library Search Path Variables" on page 18.**

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- Select the **Append** option

- If SAP is defined in Shared Services as a user directory, add

  HYPERION_HOME/common/SAP/bin

  Windows example:

  ```
  PATH  C:\Hyperion\common\SAP\bin;C:\Hyperion\common\CSS\9.5.0.0\bin
  ```

  UNIX and Linux example:

  ```
  LD_LIBRARY_PATH   /opt/hyperion/common/SAP/bin
  ```

4 **Restart the OC4J instance.**

Deploying awb.war

➤ To deploy `awb.war`:

1 **Deploy the following Web archive to the new OC4J instance:**

HYPERION_HOME/products/Foundation/BPMA/AppServer/InstallableApps/awb.war

2 **During deployment provide:**

- **Application Name**; for example, `EPMAWeb`
- Do not import the `apache.commons.logging` shared library

Applying Post Deployment Applications Settings

➤ To apply post deployment settings:

1 **Copy**

`wstx-asl-3.2.7.jar and jsr173_1.0_api.jar`

from

ORACLE_HOME/j2ee/*INSTANCE*/applications/EPMA/awb/WEB-INF/lib

to the OC4J instance library directory:

ORACLE_HOME/j2ee/*INSTANCE*/applib

2 **Restart the OC4J instance.**

Deploying Performance Management Architect DataSync

Configuring OC4J

➤ To configure OC4J:

1 **Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".**

2 **Add the Java arguments:**

-Dhyperion.home=*HYPERION_HOME*

-Djavax.xml.stream.XMLOutputFactory=com.ctc.wstx.stax.WstxOutputFactory

-Djavax.xml.stream.XMLInputFactory=com.ctc.wstx.stax.WstxInputFactory

For Solaris only:

-Djava.nio.channels.spi.SelectorProvider=sun.nio.ch.PollSelectorProvider

3 **If necessary, add a library search path variable. See "Library Search Path Variables" on page 18.**

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- Select the **Append** option
- If SAP is defined in Shared Services as a user directory, add

  HYPERION_HOME/common/SAP/bin

Windows example:

```
PATH  C:\Hyperion\common\SAP\bin;C:\Hyperion\common\CSS\9.5.0.0\bin
```

UNIX and Linux example:

```
LD_LIBRARY_PATH   /opt/hyperion/common/SAP/bin
```

4 **Restart the OC4J instance.**

Deploying DataSync.war

➤ To deploy DataSync.war:

1 **Deploy the following Web archive to the new OC4J instance:**

HYPERION_HOME/products/Foundation/BPMA/AppServer/InstallableApps/DataSync.war

2 **During deployment provide:**

- **Application Name**; for example, DataSync
- Do not import the apache.commons.logging shared library

3 **Restart the OC4J instance.**

Applying Post Deployment Applications Settings

➤ To apply post deployment settings:

1 **Copy**

`wstx-asl-3.2.7.jar and jsr173_1.0_api.jar`

from

ORACLE_HOME`/j2ee/`*INSTANCE*`/applications/EPMA/DataSync/WEB-INF/lib`

to the OC4J instance library directory:

ORACLE_HOME`/j2ee/`*INSTANCE*`/applib`

2 **Restart the OC4J instance.**

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open:**

`http://`*HOSTNAME.EXAMPLE.COM:LISTEN_PORT*`/awb/`

2 **Review the output for the OC4J instance in** *ORACLE_HOME*`/opmn/logs`.

3 **Review the product logs in** *HYPERION_HOME*`/logs/epma`.

Deploying Calculation Manager

Configuring OC4J

➤ To configure OC4J:

1 **Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".**

2 **Add the Java arguments:**

```
-Dhyperion.home=HYPERION_HOME
-noverify
```

For Solaris only:

```
-Djava.nio.channels.spi.SelectorProvider=sun.nio.ch.PollSelectorProvider
```

3 **Add the following:**

a. Add an OC4J environment variable HYPERION_HOME.

b. If necessary, add a library search path variable. See "Library Search Path Variables" on page 18.

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- Select the **Append** option
- If SAP is defined in Shared Services as a user directory, add

  HYPERION_HOME/common/SAP/bin

Windows example:

```
HYPERION_HOME   C:\Hyperion
PATH   C:\Hyperion\common\SAP\bin;C:\Hyperion\common\CSS\9.5.0.0\bin
```

UNIX and Linux example:

```
HYPERION_HOME   /opt/hyperion
LD_LIBRARY_PATH   /opt/hyperion/common/SAP/bin
```

4 **Restart the OC4J instance.**

Deploying calcmgr.ear

➤ To deploy `calcmgr.ear`:

1 **Deploy the following Enterprise archive to the OC4J instance that you created:**

HYPERION_HOME/products/Foundation/CALC/AppServer/InstallableApps/calcmgr.ear

2 **During deployment provide:**

- **Application Name**; for example, `calcmgr`
- Do not import the `apache.commons.logging` shared library
- Do not import the `oracle.xml` shared library

3 **Restart the OC4J instance.**

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open:**

http://*HOSTNAME.EXAMPLE.COM:LISTEN_PORT*/calcmgr/

2 **Review the output for the OC4J instance in** *ORACLE_HOME*/opmn/logs.

3 **Review the product logs in** *HYPERION_HOME*/logs/calcmgr.

Deploying Financial Reporting

Configuring OC4J

➤ To configure OC4J:

1 **Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".**

2 **Change the value of the `-Djava.awt.headless` Java argument from `true` to `false`.**

3 **Add the following Java argument:**

`-DHYPERION_HOME=`*`HYPERION_HOME`*

For Solaris only:

`-Djava.nio.channels.spi.SelectorProvider=sun.nio.ch.PollSelectorProvider`

4 **Add the following:**

a. Add OC4J environment variables:

- HYPERION_HOME
- ESSLANG
- ARBORPATH
- ESSBASEPATH
- ESS_ES_HOME
- **UNIX**: Add DISPLAY, and assign a valid X11 address as its value. See "Enabling an X Virtual Frame Buffer for Financial Reporting and Web Analysis" in the *Oracle Enterprise Performance Management System Installation and Configuration Guide*.

b. Add a library search path variable. See "Library Search Path Variables" on page 18.

Tip: (UNIX only): Execute: *HYPERION_HOME*`/products/biplus/InstallableApps/freporting_web.env`. Use script output as the library search path variable value.

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- Select the **Append** option.
- If SAP is used as a data source or is defined in Shared Services as a user directory, add

HYPERION_HOME/common/SAP/bin

- If Financial Management is used as a data source, add:

  HYPERION_HOME/products/FinancialManagement/Client
  HYPERION_HOME/products/FinancialManagement/common

- If Planning is used as a data source, add

  HYPERION_HOME/common/ADM/Planning/*VERSION*/bin

  where *VERSION* = 9.5.0.0 if the version of Planning is this release (11.1.1.4). If Planning is a previous release, then *VERSION* is the version of the Planning you installed.

Windows example:

```
HYPERION_HOME   C:\Hyperion
ESSLANG         English_UnitedStates.Latin1@Binary
ARBORPATH       C:\Hyperion\common\EssbaseRTC\9.5.0.0
ESSBASEPATH     C:\Hyperion\common\EssbaseRTC\9.5.0.0
ESS_ES_HOME     C:\Hyperion\products\biplus\bin\EssbaseJAPI

PATH  C:\Hyperion\products\biplus\bin;
C:\Hyperion\common\EssbaseRTC\9.5.0.0\bin;
C:\Hyperion\common\ADM\9.5.0.0\Essbase\9.5.0.0\bin;
C:\Hyperion\common\SAP\bin
```

UNIX and Linux example:

```
HYPERION_HOME   /opt/hyperion
ESSLANG         English_UnitedStates.Latin1@Binary
ARBORPATH       /opt/hyperion/common/EssbaseRTC/9.5.0.0
ESSBASEPATH     /opt/hyperion/common/EssbaseRTC/9.5.0.0
ESS_ES_HOME     /opt/hyperion/products/biplus/bin/EssbaseJAPI

DISPLAY     xvfb-host.example.com:99.0

LD_LIBRARY_PATH   /opt/hyperion/products/biplus/bin:
/opt/hyperion/common/EssbaseRTC/9.5.0.0/bin:
/opt/hyperion/common/ADM/9.5.0.0/Essbase/9.5.0.0/bin:
/opt/hyperion/common/SAP/bin
```

5 **Restart the OC4J instance.**

Deploying HReports.ear

➤ To deploy `HReports.ear`:

1 **Deploy the following enterprise archive to the OC4J instance that you created:**

HYPERION_HOME/products/biplus/InstallableApps/HReports.ear

2 **During deployment, provide:**

- **Application Name**; for example, `FinancialReporting`
- Do not import the `apache.commons.logging` shared library

3 **Restart the OC4J instance.**

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open:**

`http://HOSTNAME.EXAMPLE.COM:LISTEN_PORT/hr/status.jsp`

2 **Review the output for the OC4J instance in** *ORACLE_HOME*`/opmn/logs`.

3 **Review the product logs in** *HYPERION_HOME*`/logs/BIPlus`.

Deploying Web Analysis

Configuring OC4J

➤ To configure OC4J:

1 **Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".**

2 **Change the value of the** `-Djava.awt.headless` **Java argument from** `true` **to** `false`.

3 **Add the following Java arguments**

```
-DHYPERION_HOME=HYPERION_HOME
-DBIPLUS_HOME=HYPERION_HOME/products/biplus
```

For Solaris only:

```
-Djava.nio.channels.spi.SelectorProvider=sun.nio.ch.PollSelectorProvider
```

4 **Add the following:**

a. Add OC4J environment variables:

- HYPERION_HOME
- BIPLUS_HOME
- ESSLANG
- ARBORPATH
- ESSBASEPATH
- ESS_ES_HOME
- ICU_DATA

- **UNIX**: Add DISPLAY, and assign a valid X11 address as its value. See "Enabling an X Virtual Frame Buffer for Financial Reporting and Web Analysis" in the *Oracle Enterprise Performance Management System Installation and Configuration Guide*.

b. If necessary, add a library search path variable. See "Library Search Path Variables" on page 18.

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- Select the **Append** option.
- If SAP is used as a data source or is defined in Shared Services as a user directory, add

  HYPERION_HOME/common/SAP/bin

- If Financial Management is used as a data source, add:

  HYPERION_HOME/products/FinancialManagement/Client
  HYPERION_HOME/products/FinancialManagement/common

Windows example:

```
HYPERION_HOME C:\Hyperion
BIPLUS_HOME  C:\Hyperion\products\biplus
ESSLANG      English_UnitedStates.Latin1@Binary
ARBORPATH    C:\Hyperion\common\EssbaseRTC\9.5.0.0
ESSBASEPATH  C:\Hyperion\common\EssbaseRTC\9.5.0.0
ESS_ES_HOME  C:\Hyperion\products\biplus\bin\EssbaseJAPI
ICU_DATA     C:\Hyperion\common\ADM\9.5.0.0\Essbase\9.5.0.0\bin\HssEssDriver

PATH   C:\Hyperion\common\SAP\bin
```

UNIX and Linux example:

```
HYPERION_HOME /opt/hyperion
BIPLUS_HOME   /opt/hyperion/products/biplus
ESSLANG       English_UnitedStates.Latin1@Binary
ARBORPATH     /opt/hyperion/common/EssbaseRTC/9.5.0.0
ESSBASEPATH   /opt/hyperion/common/EssbaseRTC/9.5.0.0
ESS_ES_HOME   /opt/hyperion/products/biplus/bin/EssbaseJAPI
ICU_DATA      /opt/hyperion/common/ADM/9.5.0.0/Essbase/9.5.0.0/bin/HssEssDriver

DISPLAY     xvfb-host.example.com:99.0

LD_LIBRARY_PATH   /opt/hyperion/common/SAP/bin
```

5 **Restart the OC4J instance.**

Deploying WebAnalysis.ear

➤ To deploy `WebAnalysis.ear`:

1 **Deploy the following enterprise archive to the OC4J instance that you created:**

HYPERION_HOME/products/biplus/InstallableApps/WebAnalysis.ear

2 **During deployment, provide:**

- **Application Name**: for example, `WebAnalysis`
- Do not import the `apache.commons.logging` shared library

3 **Restart the OC4J instance.**

Configuring Web Server Routing

To improve performance and to reduce load on the application server, static content is served by the Web server.

Note: If your Web server is not on the same machine where EPM Workspace was installed, you must manually copy the following folder: *HYPERION_HOME*`/products/Foundation/workspace/AppServer/InstalledApps/WebAnalysis_static`.

Oracle HTTP Server and Apache

Note: In Apache 2.2 and later, the default access state for the Web server root directory is set to `Deny`. This setting will prevent static content for EPM Workspace and Web Analysis from displaying. Refer to the Apache Module mod_authz host documentation for information on changing the default access state.

➤ To ”configure Web server routing:

1 **If using OHS based on Apache 1.3, or Apache 1.3:**

a. Using a text editor, open `httpd.conf`.

b. Add the following directive to your Web server configuration file (`httpd.conf`), replacing *HYPERION_HOME* as needed:

```
Alias /WebAnalysis "HYPERION_HOME/products/Foundation/workspace/AppServer/
InstalledApps/WebAnalysis_static"
```

2 **If using OHS2 based on Apache 2.0, or Apache 2.*x*:**

a. Using a text editor, open `httpd.conf`.

b. Add the following directive to your Web server configuration file (`httpd.conf`), replacing *HYPERION_HOME* as needed:

```
AliasMatch /WebAnalysis/(images|js|resources|themes)/(.*) HYPERION_HOME
/products/Foundation/workspace/AppServer/InstalledApps/
WebAnalysis_static/$1/$2
```

3 **Using a text editor, open `mod_oc4j.conf` on the machine where OHS is installed:**

a. Add the following Oc4jMount directives, which instruct OHS to route requests to the specified contexts to OC4J:

```
Oc4jMount /WebAnalysis/ServerConsole/* DESTINATION
Oc4jMount /WebAnalysis/templates/* DESTINATION
Oc4jMount /WebAnalysis/hfmtemplates/* DESTINATION
```

```
Oc4jMount /WebAnalysis/hitemplates/* DESTINATION
Oc4jMount /WebAnalysis/processor DESTINATION
Oc4jMount /WebAnalysis/servlet/* DESTINATION
Oc4jMount /WebAnalysis/modules/* DESTINATION
Oc4jMount /WebAnalysis/resources/* DESTINATION
Oc4jMount /WebAnalysis/DirectoryServlet DESTINATION
Oc4jMount /WebAnalysis/config DESTINATION
Oc4jMount /WebAnalysis/wa_javadocs DESTINATION
Oc4jMount /WebAnalysis/ DESTINATION
Oc4jMount /WebAnalysis/*.jsp DESTINATION
Oc4jMount /WebAnalysis/*.xml DESTINATION
Oc4jMount /WebAnalysis/*.jar DESTINATION
Oc4jMount /WebAnalysis/*.exe DESTINATION
Oc4jMount /WebAnalysis/portletservlet/* DESTINATION
Oc4jMount /WebAnalysis/porttemplates/* DESTINATION
```

b. Replace *DESTINATION* with one of the following:

- ajp13_dest
- cluster_dest (this is the default destination type)
- instance_dest

Example:

```
Oc4jMount /WebAnalysis/ cluster://WebAnalysis
```

For information about configuring `mod_oc4j`, see the *Oracle HTTP Server Administrator's Guide*.

4 **Restart the OC4J instance and HTTP Server.**

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open:**

`http://HOSTNAME.EXAMPLE.COM:LISTEN_PORT/WebAnalysis/`

2 **Review the output for the OC4J instance in** `ORACLE_HOME/opmn/logs`

3 **Review the product logs in** `HYPERION_HOME/logs/BIPlus`.

Deploying Planning

Configuring OC4J

➤ To configure OC4J:

1 **Start the RMIRegistry for Planning using:**

- **Windows:** Windows service
- **UNIX and Linux:** Execute: *HYPERION_HOME*/common/RMI/9.5.0.0/HyperionRMIRegistry

2 **Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".**

3 **Add the Java argument:**

```
-Dhyperion.home=HYPERION_HOME
```

For Solaris only:

```
-Djava.nio.channels.spi.SelectorProvider=sun.nio.ch.PollSelectorProvider
```

4 **Add the Java argument:**

```
-Dhyperion.home=HYPERION_HOME
```

For Solaris only:

```
-Djava.nio.channels.spi.SelectorProvider=sun.nio.ch.PollSelectorProvider
```

5 **Add the following:**

a. Add OC4J environment variables:

- HYPERION_HOME
- PLANNING_HOME

b. Add a library search path variable. See "Library Search Path Variables" on page 18.

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- Select the **Append** option
- If SAP is defined in Shared Services as a user directory, add *HYPERION_HOME*/common/SAP/bin

Windows example:

```
HYPERION_HOME    C:\Hyperion
PLANNING_HOME    C:\Hyperion\products\Planning

PATH  HYPERION_HOME\common\EssbaseRTC\9.5.0.0\bin;C:\Hyperion\common\SAP\bin;
C:\Hyperion\common\CSS\9.5.0.0\bin
```

UNIX and Linux example:

```
HYPERION_HOME      /opt/Hyperion
PLANNING_HOME      /opt/Hyperion/products/Planning
```

```
LD_LIBRARY_PATH  /opt/hyperion/common/EssbaseRTC/9.5.0.0/bin:/opt/hyperion/products/
Planning/lib:/opt/hyperion/common/SAP/bin:$LD_LIBRARY_PATH";export LD_LIBRARY_PATH
```

6 **Restart the OC4J instance.**

Deploying HyperionPlanning.ear

➤ To deploy `HyperionPlanning.ear`:

1 **Deploy the following enterprise archive to the OC4J instance that you created:**

HYPERION_HOME`/products/Planning/AppServer/InstallableApps/common/HyperionPlanning.ear`

2 **During deployment provide:**

- **Application Name**; for example, `HyperionPlanning`
- Do not import the `apache.commons.logging` shared library

3 **Restart the OC4J instance.**

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open:**

`http://`*HOSTNAME.EXAMPLE.COM:LISTEN_PORT*`/HyperionPlanning/LogOn.jsp`

2 **Review the output for the OC4J instance in** *ORACLE_HOME*`/opmn/logs`.

3 Review the product logs in *HYPERION_HOME*`/logs/Planning`.

Deploying Performance Scorecard

Configuring OC4J

➤ To configure OC4J:

1 **Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".**

2 **Add the following Java arguments:**

```
-Dhyperion.home=HYPERION_HOME
-DESS_ES_HOME=HYPERION_HOME/common/EssbaseRTC/9.5.0.0
```

For Solaris only:

```
-Djava.nio.channels.spi.SelectorProvider=sun.nio.ch.PollSelectorProvider
```

3 **Add the following:**

a. Add OC4J environment variables:

- HYPERION_HOME
- ARBORPATH
- ESSBASEPATH
- ESSLANG

b. Add a library search path variable. See "Library Search Path Variables" on page 18.

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- Select the **Append** option
- If SAP is defined in Shared Services as a user directory, add

  HYPERION_HOME/common/SAP/bin

Windows example:

```
HYPERION_HOME   C:\Hyperion
ARBORPATH       C:\Hyperion\common\EssbaseRTC\9.5.0.0
ESSBASEPATH     C:\Hyperion\common\EssbaseRTC\9.5.0.0
ESSLANG         English_UnitedStates.Latin1@Binary

PATH  C:\Hyperion\common\EssbaseRTC\9.5.0.0\bin;
C:\Hyperion\common\ADM\9.5.0.0\Essbase\9.5.0.0\bin;
C:\Hyperion\common\CSS\9.5.0.0\bin
```

UNIX and Linux example:

```
HYPERION_HOME   /opt/hyperion
ARBORPATH   /opt/hyperion/common/EssbaseRTC/9.5.0.0
ESSBASEPATH   /opt/hyperion/common/EssbaseRTC/9.5.0.0
ESSLANG   English_UnitedStates.Latin1@Binary

LD_LIBRARY_PATH  /opt/common/EssbaseRTC/9.5.0.0/bin:
opt/common/ADM/9.5.0.0/Essbase/9.5.0.0/bin:
/opt/hyperion/common/SAP/bin
```

4 **Create an OC4J Shared Library that includes**

HYPERION_HOME/products/PerformanceScorecard/AppServer/InstallableApps/common/webappsconf/lib/xml/xercesImpl.jar.

During Shared Library creation, provide:

- **Shared Library Name**; for example, `apache.xml`
- **Shared Library Version**; for example, `2.5`

5 **Restart the OC4J instance.**

Deploying HPSWebReports.war

➤ To deploy `HPSWebReports.war`:

1 **Deploy the following Web archive to the new OC4J instance:**

HYPERION_HOME`/products/PerformanceScorecard/AppServer/InstallableApps/common/webapps/configured/HPSWebReports.war`

2 **During deployment provide:**

- **Application Name**; for example, `HPS`
- Do not import the `apache.commons.logging` shared library
- Do not import the `oracle.xml` shared library
- Import the shared library that you created; for example, `apache.xml`

Deploying HPSAlerter.war

➤ To deploy `HPSAlerter.war`:

1 **Deploy the following Web archive to the new OC4J instance:**

HYPERION_HOME`/products/PerformanceScorecard/AppServer/InstallableApps/common/webapps/configured/HPSAlerter.war`

2 **During deployment provide:**

- **Application Name**; for example, `HPSAlerter`
- Do not import the `apache.commons.logging` shared library
- Do not import the oracle.xml shared library
- Import the shared library that you created; for example, `apache.xml`

3 **Restart the OC4J instance.**

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open:**

`http://`*HOSTNAME.EXAMPLE.COM:LISTEN_PORT*`/workspace/`

2 **Access the Scorecard module using the Workspace navigation menu**

3 **Review the output for the OC4J instance in** *ORACLE_HOME*`/opmn/logs`.

4 **Review the product logs in** *HYPERION_HOME*`/logs/hps`.

Deploying Profitability and Cost Management

Configuring OC4J

➤ To configure OC4J:

1 **Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".**

2 **Add the Java argument:**

`-Dhyperion.home=`*`HYPERION_HOME`*

For Solaris only:

`-Djava.nio.channels.spi.SelectorProvider=sun.nio.ch.PollSelectorProvider`

3 **Add an OC4J environment variable HYPERION_HOME.**

Windows example:

```
HYPERION_HOME    C:\Hyperion
```

UNIX and Linux example:

```
HYPERION_HOME    /opt/hyperion
```

4 **Restart the OC4J instance.**

Deploying profitability.war

➤ To deploy `profitability.war`:

1 **Deploy the following Web archive to the OC4J instance that you created:**

`HYPERION_HOME``/products/Profitability/AppServer/InstallableApps/common/profitability.war`

2 **During deployment provide:**

- **Application Name**: For example: `Profitability`
- **Parent Name**: Select **default**
- **Bind Web Module to Site**: Select **default-web-site**
- **Context Root**: Enter **profitability**
- Do not import the `apache.commons.logging` shared library
- Do not import the `oracle.persistence` shared library
- Import the shared library that you created; for example: `oracle.epm.persistence`

3 **Restart the OC4J instance.**

Post Deployment Configurations

Adding Parameters

➤ After deploying Profitability and Cost Management to the Oracle Application Server, you must:

1 **Add the Java argument -userThreads.**

2 **Restart the OC4J instance.**

Configuring the Shared Library on Oracle Application Server

After installing and configuring Profitability and Cost Management, you must perform the following procedure to include the DataDirect drivers in the `oracle.persistence` library. This step is necessary to ensure you can properly create import configurations for staging tables.

➤ To configure the Shared Library:

1 **After manually deploying Profitability to the Oracle Application Server (OAS), navigate to the** `profitability.war` **file in**

ORACLE_HOME`\product\10.1.3.1\OracleAS_1\j2ee\Profitability\applications\Profitability`

Note: By default, if the application is deployed automatically, the `profitability.war` file is located in `Hyperion\deployment\Oracle10g\Profitability\applications\Profitability\profitability`

2 **From** `profitability.war`**, extract** `\WEB-INF\lib\hyjdbc.jar`

3 **Copy the file to** *ORACLE_HOME*`\toplink\jlib\`.

4 **In *ORACLE_HOME*\j2ee*INSTANCE*\config\server.xml file, locate the following code:**

```
<shared-library name="oracle.persistence" version="1.0" librarycompatible="true">
      <code-source path="../../../toplink/jlib/toplink-essentials.jar"/>
      <import-shared-library name="oracle.jdbc"/>
</shared-library>
```

5 **Add the reference to** `hyjdbc.jar`**, as shown in bold in the following example:**

```
<shared-library name="oracle.persistence" version="1.0" librarycompatible="true">
      <code-source path="../../../toplink/jlib/toplink-essentials.jar"/>
      <code-source path="../../../toplink/jlib/hyjdbc.jar"/>
      <import-shared-library name="oracle.jdbc"/>
</shared-library>
```

6 **Restart Profitability and Cost Management.**

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open:**

`http://HOSTNAME.EXAMPLE.COM:LISTEN_PORT/Profitability/ping.jsp`

2 **Review the output for the OC4J instance in** *ORACLE_HOME*`/opmn/logs`.

3 **Review the product logs in** *HYPERION_HOME*`/logs/Profitability`.

Configuring Financial Management With a Web Server

Apache Web Server or Oracle HTTP Server

To configure Financial Management with Apache Web Server or Oracle HTTP Server: using a text editor, open the Web server's configuration file (`httpd.conf`) and add the following directives, replacing *HFM_HOST* and *HFM_PORT* with real values:

```
#Change modules/mod_proxy.so as needed; e.g. libexec/mod_proxy on UNIX
<IfModule !mod_proxy.c>
LoadModule proxy_module modules/mod_proxy.so
</IfModule>
<IfModule !mod_proxy_connect.c>
LoadModule proxy_connect_module modules/mod_proxy_connect.so
</IfModule>
<IfModule !mod_proxy_http.c>
LoadModule proxy_http_module modules/mod_proxy_http.so
</IfModule>
ProxyRequests Off
ProxyPass /hfm http://HFM_HOST:HFM_PORT/hfm
ProxyPassReverse /hfm http://HFM_HOST:HFM_PORT/hfm
```

where *HFM_HOST* is the Financial Management host machine and *HFM_PORT* is the default port, 80.

Using Only IIS with Financial Management

For using only IIS (for example, if Apache is not used at all): You must configure the IIS used by Financial Management as the front-end Web server for EPM Workspace. You can have the EPM Workspace Web application server running on a separate machine, but you must install a copy of the EPM Workspace component onto the machine running IIS. The EPM Workspace installation includes files that are served by the Web server, such as the help files.

After installing EPM Workspace on the IIS machine, use Life Cycle Management to change the Web server hostname.

Configuring BI Publisher with Apache Web Server or Oracle HTTP Server

To configure Oracle Business Intelligence Publisher with Apache Web Server or Oracle HTTP Server: using a text editor, open the Web server's configuration file (`httpd.conf`), and add the following directives, replacing *`BIPUB_HOST`* and *`BIPUB_PORT`* with real values:

```
 #Change modules/mod_proxy.so as needed; e.g. libexec/mod_proxy on UNIX
<IfModule !mod_proxy.c>
LoadModule proxy_module modules/mod_proxy.so
</IfModule>
<IfModule !mod_proxy_connect.c>
LoadModule proxy_connect_module modules/mod_proxy_connect.so
</IfModule>
<IfModule !mod_proxy_http.c>
LoadModule proxy_http_module modules/mod_proxy_http.so
</IfModule>
ProxyRequests Off
ProxyPass /xmlpserver http://BIPUB_HOST:BIPUB_PORT/xmlpserver
ProxyPassReverse /xmlpserver http://BIPUB_HOST:BIPUB_PORT/xmlpserver
```

Configuring Oracle BI EE with Apache Web Server or Oracle HTTP Server

To configure Oracle Business Intelligence Enterprise Edition with Apache Web Server or Oracle HTTP Server: using a text editor, open the Web server's configuration file (`httpd.conf`), and add the following directives, replacing *`OBIEE_HOST`* and *`OBIEE_PORT`* with real values:

```
 #Change modules/mod_proxy.so as needed; e.g. libexec/mod_proxy on UNIX
<IfModule !mod_proxy.c>
LoadModule proxy_module modules/mod_proxy.so
</IfModule>
<IfModule !mod_proxy_connect.c>
LoadModule proxy_connect_module modules/mod_proxy_connect.so
</IfModule>
<IfModule !mod_proxy_http.c>
LoadModule proxy_http_module modules/mod_proxy_http.so
</IfModule>
ProxyRequests Off
ProxyPass /analytics http://OBIEE_HOST:OBIEE_PORT/analytics
ProxyPassReverse /analytics http://OBIEE_HOST:OBIEE_PORT/analytics
```

4 Deploying EPM System into WebLogic Server

In This Chapter

WebLogic Server supports several distinctively different methods for controlling managed servers in a domain. The instructions that follow assume the Java based Node Manager is used, though the information, in general, can be applied to all supported configurations. See *WebLogic Server Managing Server Startup and Shutdown*.

WebLogic Prerequisites

- Identify an existing, or create a new, WebLogic domain for use by EPM System
- Install a WebLogic supported Web server, and WebLogic Web server plug-ins. See "Configuring Web Server Routing" on page 91.

Tip: Apache HTTP Server is included with the EPM System, and is installed in *HYPERION_HOME*/common/httpServers/Apache.

- Install and configure BEA Java based Node Manager on each machine that will host an EPM System Web application.

  This guide assumes that EPM System Web application settings such as Java arguments and CLASSPATH are defined using the Start Server configuration page in the WebLogic administration console for the managed server on which you are deploying.

WebLogic Node Manager

WebLogic's Java-based Node Manager allows you to start, stop, suspend, and restart managed server instances in your domain. Additionally, it can monitor the health of your managed servers, and restart them automatically in the event of failure.

Enabling Node Manager Start Script

When using script-based managed server control, environment variable definitions can be included in the start script for each managed server. Similarly, using WebLogic's Java-based Node Manager, you can use a custom start script to initialize environment variables before starting a managed server. To enable this feature, the following properties must be defined in `nodemanager.properties` for each Node Manager instance: `StartScriptEnabled`, `StartScriptName`.

➤ To enable Node Manager start script, for each instance of Node Manager in the domain:

1 **Open or create:**

WL_HOME`/common/nodemanager/nodemanager.properties`

2 **Add the following properties:**

```
StartScriptEnabled=true
StartScriptName=startEPMSystem.cmd|sh
```

3 **Restart Node Manager and administrator console.**

Creating Node Manager Start Script

Create

DOMAIN_HOME`/bin/startEPMSystem.cmd|sh`

substituting *DOMAIN_HOME* with the file system path of the WebLogic domain:

Windows:

```
@echo off

rem startEPMSystem.cmd
rem Custom Node Manager start script to set EPM System specific parameters

rem This release of EPM System supports one HYPERION_HOME.
rem Change this value to match your HYPERION_HOME location, if different.
set HYPERION_HOME=C:\Hyperion
```

```
rem setDomainEnv.cmd will reset CLASSPATH, but EXT_PRE_CLASSPATH is prepended.
set EXT_PRE_CLASSPATH=%CLASSPATH%

rem Node Manager sets %SERVER_NAME%
goto %SERVER_NAME%

:STARTWLS
%~dp0/startManagedWebLogic.cmd
exit
```

UNIX and Linux:

```
#!/bin/sh

# startEPMSystem.sh
# Custom Node Manager start script to set EPM System specific parameters

# This release of EPM System supports one HYPERION_HOME.
HYPERION_HOME=HYPERION_HOME
export HYPERION_HOME

# Node Manager sets path to "" on UNIX/Linux
PATH=/bin:/usr/bin:${PATH}
export PATH

# setDomainEnv.sh will reset CLASSPATH, but EXT_PRE_CLASSPATH is prepended.
EXT_PRE_CLASSPATH=${CLASSPATH}
export EXT_PRE_CLASSPATH

# Node Manager sets ${SERVER_NAME}
case ${SERVER_NAME} in
    * ) ;;
esac

`dirname ${0}`/startManagedWebLogic.sh
```

Evaluating SERVER_NAME in Start Script

As managed servers are added to the domain, conditional statements are added to the script that control the flow of execution based on SERVER_NAME, which is set by the Node Manager. For example, the following script defines a variable `FOO` for the managed server `Server0`. `FOO` will not be set for other managed servers.

Windows example:

```
@echo off

rem startEPMSystem.cmd
rem Custom Node Manager start script to set EPM System specific parameters

rem This release of EPM System supports one HYPERION_HOME.
rem Change this value to match your HYPERION_HOME location, if different.
set HYPERION_HOME=C:\Hyperion

rem setDomainEnv.cmd will reset CLASSPATH, but EXT_PRE_CLASSPATH is prepended.
```

```
set EXT_PRE_CLASSPATH=%CLASSPATH%

rem Node Manager sets %SERVER_NAME%
goto %SERVER_NAME%

:STARTWLS
%~dp0/startManagedWebLogic.cmd
exit

:Server0
set FOO=BAR
goto STARTWLS
```

UNIX and Linux:

```
#!/bin/sh

# startEPMSystem.sh
# Custom Node Manager start script to set EPM System specific parameters

# This release of EPM System supports one HYPERION_HOME.
HYPERION_HOME=HYPERION_HOME
export HYPERION_HOME

# Node Manager sets path to "" on UNIX/Linux
PATH=/bin:/usr/bin:${PATH}
export PATH

# setDomainEnv.sh will reset CLASSPATH, but EXT_PRE_CLASSPATH is prepended.
EXT_PRE_CLASSPATH=${CLASSPATH}
export EXT_PRE_CLASSPATH

# Node Manager sets ${SERVER_NAME}
case ${SERVER_NAME} in
    Server0 )
        FOO=BAR
        export FOO
        ;;
    * ) ;;
esac

`dirname ${0}`/startWebLogic.sh
```

Deploying Shared Services

Configuring Managed Server

Note: This document assumes your managed server is named `SharedServices9`.

➤ To configure managed server:

1 **Start OpenLDAP using the Windows service, or by running**

HYPERION_HOME/products/Foundation/openLDAP/startService.bat|startOpenLDAP.sh

2 **Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".**

3 **Add the following Java arguments to the Server Start configuration for the managed server:**

```
-DHYPERION_HOME=HYPERION_HOME
-Dhyperion.home=HYPERION_HOME
```

4 **Add the following CLASSPATH entries to the Server Start configuration for the managed server:**

```
HYPERION_HOME/deployments/WebLogic9/SharedServices9/config
HYPERION_HOME/common/JakartaCommons/commons-dbcp-1.2.1.jar
HYPERION_HOME/common/JDBC/DataDirect/3.7/lib/hyjdbc.jar
HYPERION_HOME/common/JakartaCommons/commons-pool-1.3.jar
HYPERION_HOME/common/SAP/lib
```

Note: Separate entries with semi-colons (;) for Windows, colons (:) for UNIX.

5 **Add a condition to**

DOMAIN_HOME/bin/startEPMSystem.cmd|sh

See "Evaluating SERVER_NAME in Start Script" on page 57.

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- If SAP is defined in Shared Services as a user directory, add

  HYPERION_HOME/common/SAP/bin

Windows example:

```
:SharedServices
call %HYPERION_HOME%\products\Foundation\OpenLDAP\startService.bat
set PATH=%HYPERION_HOME%\common\CSS\9.5.0.0\bin
set PATH=%PATH%;%HYPERION_HOME%\common\SAP\bin
goto STARTWLS
```

UNIX and Linux example:

```
case ${SERVER_NAME} in
SharedServices )
    ${HYPERION_HOME}/products/Foundation/openLDAP/startOpenLDAP.sh
    LD_LIBRARY_PATH=$HYPERION_HOME/common/SAP/bin:${LD_LIBRARY_PATH};     export
LD_LIBRARY_PATH
    ;;
* ) ;;
```

6 **To prevent WebLogic from trying to authenticate basic authentication headers, set the `enforce-valid-basic-auth-credentials` element to false for your Weblogic domain. Refer to the "Understanding BASIC Authentication with Unsecured Resources" section of the *BEA's Programming WebLogic Security Guide* for information on this setting. For example, in your domain's `config.xml`, add the element within the `<security-configuration>`.**

For example:

```
<security-configuration>
   <enforce-valid-basic-auth-credentials>false</enforce-valid-basic-auth-
credentials>
</security-configuration>
```

7 **Restart the managed server.**

Deploying interop.war

➤ To deploy `interop.war`:

1 **Deploy the following Web archive directory to the managed server that you created:**

HYPERION_HOME`/deployments/WebLogic9/servers/SharedServices9/webapps/interop`

2 **During deployment, provide:**

- Select **Install this deployment as an application**
- Select the managed server you created as the application deployment target; for example, `SharedServices`
- **Name**: for example, **`SharedServices`**
- Select **I will make the deployment accessible from the following location**, and ensure the pre-filled file system location for the expanded archive is correct
- Start servicing all requests for the deployed application, switching **State** from **Prepared** to **Active**

3 **To access the deployed application through a Web server, see "Configuring Web Server Routing" on page 91.**

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open:**

`http://`*HOSTNAME.EXAMPLE.COM:LISTEN_PORT*`/interop/`

2 **Log on to Shared Services.**

3 **Review the output for your WebLogic managed server in** *DOMAIN_HOME*`/servers/`*SERVER_NAME*`/logs`.

4 **Review the product logs in** *HYPERION_HOME*`/logs/SharedServices9`.

Deploying Administration Services

- "Configuring Managed Server" on page 61
- "Deploying eas.ear" on page 62

- "Verifying Deployment" on page 63

Configuring Managed Server

Note: This document assumes your managed server is named EAS.

➤ To configure managed server:

1 **Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".**

2 **Add the following Java arguments to the Server Start configuration for the managed server:**

```
-Dhyperion.home=HYPERION_HOME
-DHYPERION_HOME=/HYPERION_HOME
-DESS_ES_HOME=/HYPERION_HOME/products/Essbase/eas/server
-DEAS_HOME=HYPERION_HOME/products/Essbase/eas
-DEAS_LOG_LEVEL=5000
-DEAS_LOG_LOCATION=HYPERION_HOME/logs/eas/easserver.log
-DEAS_SERVER_VERSION=11.1.1.4
```

3 **Add the following CLASSPATH entry to the Server Start configuration for the managed server:**

```
HYPERION_HOME/common/JakartaCommons/commons-lang-2.1.jar
```

4 **Add the following:**

a. Add a condition to

```
DOMAIN_HOME/bin/startEPMSystem.cmd|sh
```

that defines following environment variables for the managed server:

- EAS_HOME
- ESSLANG
- ARBORPATH
- ESSBASEPATH

See "Evaluating SERVER_NAME in Start Script" on page 57.

b. Add a library search path variable. See "Library Search Path Variables" on page 18.

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- If SAP is defined in Shared Services as a user directory, add

```
HYPERION_HOME/common/SAP/bin
```

Windows example:

```
:EAS
set EAS_HOME=%HYPERION_HOME%\products\Essbase\eas
set ESSLANG=English_UnitedStates.Latin1@Binary
set ARBORPATH=%EAS_HOME%\server
```

```
set ESSBASEPATH=%ARBORPATH%
set PATH=%EAS_HOME%\server\bin
set PATH=%PATH%;%HYPERION_HOME%\common\SAP\bin
set PATH=%PATH%;%HYPERION_HOME%\common\CSS\9.5.0.0\bin
goto STARTWLS
```

UNIX and Linux example:

```
EAS )
  EAS_HOME=${HYPERION_HOME}/products/Essbase/eas
  export EAS_HOME
  ESSLANG=English_UnitedStates.Latin1@Binary
  export ESSLANG
  ARBORPATH=${EAS_HOME}/server
  export ARBORPATH
  ESSBASEPATH=${ARBORPATH}
  export ESSBASEPATH
  LD_LIBRARY_PATH=${EAS_HOME}/server/bin:${HYPERION_HOME}/SAP/bin:${LD_LIBRARY_PATH}
  export LD_LIBRARY_PATH
  ;;
```

5 **Restart the managed server.**

Deploying eas.ear

➤ To deploy `eas.ear`:

1 **Create a script to expand the following Enterprise archive, and the Web archives contained within, to a file system accessible to your WebLogic domain:**

HYPERION_HOME/products/Essbase/eas/server/AppServer/InstallableApps/common/eas.ear

Windows script example:

```
set PATH=C:\bea\jdk150_10\bin;%PATH%
mkdir C:\Hyperion\deployments\WebLogic9\servers\EAS\webapps\eas
cd /d C:\Hyperion\deployments\WebLogic9\servers\EAS\webapps\eas
jar -xf C:\Hyperion\products\Essbase\eas\server\AppServer\InstallableApps\Common
\eas.ear eas.war hbrlauncher.war easdocs.war easconsole.war
mkdir eas easconsole easdocs hbrlauncher
cd eas
jar -xf ..\eas.war
cd ..\easconsole
jar -xf ..\easconsole.war
cd ..\easdocs
jar -xf ..\easdocs.war
cd ..\hbrlauncher
jar -xf ..\hbrlauncher.war
del ..\*.war
pause
```

UNIX and Linux script example:

```
#!/bin/sh
PATH=/opt/bea/jdk150_10/bin:${PATH} ; export PATH
mkdir -p /opt/hyperion/deployments/WebLogic9/servers/EAS/webapps/eas
cd /opt/hyperion/deployments/WebLogic9/servers/EAS/webapps/eas
```

```
jar -xf /opt/hyperion/products/Essbase/eas/server/AppServer/InstallableApps/common/
eas.ear
mkdir eas easconsole easdocs hbrlauncher
cd eas && jar -xf ../eas.war
cd ../easconsole && jar -xf ../easconsole.war
cd ../easdocs && jar -xf ../easdocs.war
cd ../hbrlauncher && jar -xf ../hbrlauncher.war
rm ../*.war && cd ../
```

2 **Deploy the resultant `eas` folder to the newly created managed server.**

Note: Deploying the expanded Web archive enables editing of runtime descriptor elements without requiring redeployment, and is required to access the Java Web Start version of EAS Console.

3 **During deployment, provide:**

- Select **Install this deployment as an application**
- Select the managed server you created as the application deployment target; for example, `EAS`
- Select **I will make the deployment accessible from the following location**, and ensure the pre-filled file system location for the expanded archive is correct
- Start servicing all requests for the deployed application, switching **State** from **Prepared** to **Active**

4 **To access the deployed application through a Web server, see "Configuring Web Server Routing" on page 91.**

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open:**

`http://`*`HOSTNAME.EXAMPLE.COM:LISTEN_PORT`*`/easconsole/console.html`

2 **Log on to Administration Services using the Java Web Start console.**

3 **Review the output for your WebLogic managed server in** *`DOMAIN_HOME`*`/servers/`*`SERVER_NAME`*`/logs`**.**

4 **Review the product logs in** *`HYPERION_HOME`*`/logs/eas`**.**

Deploying Provider Services

Configuring Managed Server

Note: This document assumes your managed server is named `APS`.

➤ To configure managed server:

1 **Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".**

2 **Add the following Java argument to the Server Start configuration for the managed server:**

`-DESS_ES_HOME=`*`HYPERION_HOME`*`/products/Essbase/aps`

3 **Add a library search path variable. See "Library Search Path Variables" on page 18.**

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- If SAP is defined in Shared Services as a user directory, add

  `HYPERION_HOME``/common/SAP/bin`

Windows example:

```
:APS
set PATH=%HYPERION_HOME%\products\ESSBASE\aps\bin
set PATH =%PATH%;%HYPERION_HOME%\common\SAP\bin
set PATH=%PATH%;%HYPERION_HOME%\common\CSS\9.5.0.0\bin
goto STARTWLS
```

UNIX and Linux example:

```
APS )
ESSBASEPATH=${HYPERION_HOME}/products/Essbase/aps
export ESSBASEPATH
PATH=${HYPERION_HOME}/products/Essbase/aps/bin:${PATH}
export PATH
LD_LIBRARY_PATH=${HYPERION_HOME}/common/SAP/bin:${LD_LIBRARY_PATH}
export LD_LIBRARY_PATH
;;
```

Deploying aps.war

➤ To deploy `aps.war`:

1 **Create a script to expand the following Web archive to a file system accessible to your WebLogic domain:**

`HYPERION_HOME``/products/Essbase/aps/server/AppServer/InstallableApps/common/aps.war`

Windows script example:

```
set PATH=C:\bea\jdk150_10\bin;%PATH%
mkdir C:\Hyperion\deployments\WebLogic9\servers\APS\webapps\aps
cd /d C:\Hyperion\deployments\WebLogic9\servers\APS\webapps\aps
```

```
jar -xf C:\Hyperion\products\Essbase\aps\AppServer\InstallableApps\Common\aps.war
pause
```

UNIX and Linux script example:

```
#!/bin/sh
PATH=/opt/bea/jdk150_10/bin:${PATH} ; export PATH
mkdir -p /opt/hyperion/deployments/WebLogic9/servers/APS/webapps/aps
cd /opt/hyperion/deployments/WebLogic9/servers/APS/webapps/aps
jar -xf /opt/hyperion/products/Essbase/aps/AppServer/InstallableApps/common/aps.war
```

2 **Deploy the resultant `aps` folder to the newly created managed server.**

Note: Deploying the expanded Web archive enables editing of runtime descriptor elements without requiring redeployment.

3 **During deployment, provide:**

- Select **Install this deployment as an application**
- Select the managed server you created as the application deployment target; for example, `APS`
- Select **I will make the deployment accessible from the following location**, and ensure the pre-filled file system location for the expanded archive is correct
- Start servicing all requests for the deployed application, switching **State** from **Prepared** to **Active**

4 **To access the deployed application through a Web server, see** **"Configuring Web Server Routing" on page 91**.

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open:**

`http://`*`HOSTNAME.EXAMPLE.COM:LISTEN_PORT`*`/aps/APS`

2 **Review the output for your WebLogic managed server in** *`DOMAIN_HOME`*`/servers/`*`SERVER_NAME`*`/logs`.

3 **Review the product logs in** *`HYPERION_HOME`*`/logs/aps`.

Deploying EPM Workspace

Configuring Managed Server

Note: This document assumes your managed server is named `Workspace`.

➤ To configure managed server:

1 **Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".**

2 **Add the following Java arguments to the Server Start configuration for the managed server:**

```
-Dhyperion.home=HYPERION_HOME
-Djavax.xml.parsers.SAXParserFactory=org.apache.xerces.jaxp.SAXParserFactoryImpl
```

3 **Add the following CLASSPATH entries to the Server Start configuration for the managed server:**

```
HYPERION_HOME/products/Foundation/workspace/lib/iona63.jar
HYPERION_HOME/products/Foundation/workspace/lib
HYPERION_HOME/common/SAP/lib
```

Note: Separate entries with semi-colons (;) for Windows, colons (:) for UNIX.

4 **Add a condition to `startEPMSsystem.cmd|sh`. See "Evaluating SERVER_NAME in Start Script" on page 57.**

5 **If necessary, add a library search path variable. See "Library Search Path Variables" on page 18.**

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- If SAP is defined in Shared Services as a user directory, add

  `HYPERION_HOME/common/SAP/bin`

Windows example:

```
:Workspace
set PATH=%PATH%;%HYPERION_HOME%\common\SAP\bin
set PATH=%HYPERION_HOME%\common\CSS\9.5.0.0\bin
goto STARTWLS
```

UNIX and Linux example:

```
Workspace )
  LD_LIBRARY_PATH=${HYPERION_HOME}/common/SAP/bin:${LD_LIBRARY_PATH}
  export LD_LIBRARY_PATH
  ;;
```

6 **Restart the managed server.**

Deploying workspace.war

➤ To deploy `workspace.war`:

1 **Create a script to expand the following Web archive to a file system accessible to the WebLogic domain:**

HYPERION_HOME/products/Foundation/workspace/InstallableApps/worskpace.war

Windows example:

```
set PATH=C:\bea\jdk150_10\bin;%PATH%
mkdir C:\Hyperion\deployments\WebLogic9\servers\Workspace\webapps\workspace
cd /d C:\Hyperion\deployments\WebLogic9\servers\Workspace\webapps\workspace
jar -xf C:\Hyperion\products\Foundation\workspace\InstallableApps\workspace.war
pause
```

UNIX and Linux example:

```
PATH=/opt/bea/jdk150_10/bin:${PATH}; export PATH
mkdir -p /opt/hyperion/deployments/WebLogic9/servers/Workspace/webapps/workspace
cd /opt/hyperion/deployments/WebLogic9/servers/Workspace/webapps/workspace
jar -xf /opt/hyperion/products/Foundation/workspace/InstallableApps/workspace.war
```

2 **Deploy to resultant `workspace` folder to the newly created managed server.**

Note: Deploying the expanded Web archive enables editing of runtime descriptor elements without requiring redeployment.

3 **During deployment, provide:**

- Select **Install this deployment as an application**
- Select the managed server you created as the application deployment target; for example, `Workspace`
- Select **I will make the deployment accessible from the following location**, and ensure the pre-filled file system location for the expanded archive is correct
- Start servicing all requests for the deployed application, switching **State** from **Prepared** to **Active**

4 **Perform the steps in** "Configuring Web Server Routing" on page 91.

Optional: Deploying biplus_webservices.war

➤ To deploy `biplus_webservices.war`:

1 **Deploy the following Web archive to the instance that you created:**

HYPERION_HOME/products/Foundation/workspace/InstallableApps/biplus_webservices.war

2 **During deployment, provide the Application Name; for example,** `BIPlusWebServices`.

3 **Restart the instance.**

Applying Post Deployment Application Settings

➤ To apply post deployment settings:

1 **Update:**

HYPERION_HOME\deployments\WebLogic9\servers\Workspace\webapps\workspace\WEB-INF\weblogic.xml

2 **Locate the container-descriptor tag and insert the parameter:**

```
<container-descriptor>
    <servlet-reload-check-secs>-1</servlet-reload-check-secs>
</container-descriptor>
```

Add jsp-param before the </jsp-descriptor> tag:

```
<jsp-param>
        <param-name>pageCheckSeconds</param-name>
        <param-value>-1</param-value>
</jsp-param>
```

3 **Optional: To improve performance, precompile JSP pages by creating and executing** DOMAIN_HOME/precompile.bat|sh.

Windows example:

```
echo off
SET HYPERION_HOME=C:\Hyperion
SET DOMAIN_HOME=C:\Hyperion\deployments\WebLogic9

call %DOMAIN_HOME%\bin\setDomainEnv.cmd
@rem PRECOMPILE_DIR - holds value identical to workingDir JSP parameter in
weblogic.xml
SET PRECOMPILE_DIR=.\precompiled\workspace
SET WEB_APP=%HYPERION_HOME%\deployments\WebLogic9\servers\Workspace\webapps
\workspace
echo %WEB_APP%

SET CLASSPATH=%ARDIR%\weblogic.jar;%JAVA_HOME%\jre\lib\rt.jar;%JAVA_HOME%\lib
\tools.jar;%DOMAIN_HOME%\%PRECOMPILE_DIR%;%CLASSPATH%
%JAVA_HOME%\bin\java weblogic.jspc -compilerclass com.sun.tools.javac.Main -d
%DOMAIN_HOME%\%PRECOMPILE_DIR% -compileAll -webApp %WEB_APP%
```

UNIX and Linux example:

```
#!/bin/sh
JAVA_HOME=/opt/bea/jdk150_10
HYPERION_HOME=/opt/hyperion
DOMAIN_HOME=/opt/hyperion/deployments/WebLogic9
. $DOMAIN_HOME/bin/setDomainEnv.sh
# PRECOMPILE_DIR - holds value identical to workingDir JSP parameter in weblogic.xml
PRECOMPILE_DIR=./precompiled/workspace
WEB_APP=$HYPERION_HOME/deployments/WebLogic9/servers/Workspace/webapps/workspace
CLASSPATH=$ARDIR/weblogic.jar:$JAVA_HOME/jre/lib/rt.jar:$JAVA_HOME/lib/tools.jar:
$DOMAIN_HOME/$PRECOMPILE_DIR:${CLASSPATH%}
$JAVA_HOME/bin/java weblogic.jspc -compilerclass com.sun.tools.javac.Main -d
$DOMAIN_HOME/$PRECOMPILE_DIR -compileAll -webApp $WEB_APP
```

Customizing EPM Workspace Services Configuration Scripts

EPM Workspace Services include scripts that can be launched interactively to configure various part of the system. When the Manual option is selected during EPM Workspace deployment, the *DEPLOYMENT_HOME* variable declarations must be manually defined in

HYPERION_HOME/products/Foundation/workspace/bin/settrustedpass.bat|sh

➤ To declare the variable declarations:

1 **In a text editor, open:**

HYPERION_HOME/products/Foundation/workspace/bin/settrustedpass.bat|sh

2 **Replace occurrences of the *DEPLOYMENT_HOME* with:**

$J(trustedPass.deploymentHome)

where *DEPLOYMENT_HOME* is the file system path to the deployed EPM Workspace Web application.

Windows example:

set DEPLOYMENT_HOME=*DOMAIN_HOME*/servers/Workspace/webapps/workspace

UNIX and Linux example:

DEPLOYMENT_HOME=DOMAIN_HOME/servers/Workspace/webapps/workspace

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open**

http://*HOSTNAME.EXAMPLE.COM:LISTEN_PORT*/workspace/

2 **Review the output for your WebLogic managed server in** *DOMAIN_HOME*/servers/*SERVER_NAME*/logs.

3 **Review the product log:**

HYPERION_HOME/logs/workspace

Deploying Performance Management Architect

Deploying Performance Management Architect Web

Configuring Managed Server

Note: This document assumes your managed server is named EPMAWeb

➤ To configure managed server:

1 Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".

2 Add the following Java argument to the Server Start configuration for the managed server:

`-Dhyperion.home=`*`HYPERION_HOME`*

3 Add a condition to `startEPMSsystem.cmd|sh`. See "Evaluating SERVER_NAME in Start Script" on page 57.

4 If necessary, add a library search path variable. See "Library Search Path Variables" on page 18.

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- If SAP is defined in Shared Services as a user directory, add

  `HYPERION_HOME``/common/SAP/bin`

Windows example:

```
:EPMAWeb
set PATH=%PATH%;%HYPERION_HOME%\common\SAP\bin
set PATH=%HYPERION_HOME%\common\CSS\9.5.0.0\bin
goto STARTWLS
```

UNIX and Linux example:

```
EPMAWeb )
  LD_LIBRARY_PATH=${HYPERION_HOME}/common/SAP/bin:${LD_LIBRARY_PATH}
  export LD_LIBRARY_PATH
  ;;
```

5 Restart the managed server.

Deploying awb.war

➤ To deploy `awb.war`:

1 Create a script to expand the following Web archive to a file system accessible to the WebLogic domain:

`HYPERION_HOME``/products/Foundation/BPMA/AppServer/InstallableApps/awb.war`

Windows script example:

```
set PATH=C:\bea\jdk150_10\bin;%PATH% mkdir C:\Hyperion\deployments\WebLogic9\servers
\BPMA\webapps\awb
cd C:\Hyperion\deployments\WebLogic9\servers\BPMA\webapps\awb
jar -xf C:\Hyperion\products\Foundation\BPMA\AppServer\InstallableApps\awb.war
pause
```

UNIX and Linux script example:

```
PATH=/opt/bea/jdk150_10/bin:${PATH} ; export PATH
mkdir -p /opt/hyperion/deployments/WebLogic9/servers/BPMA/webapps/awb
cd /opt/hyperion/deployments/WebLogic9/servers/BPMA/webapps/awb
jar -xf /opt/hyperion/products/Foundation/BPMA/AppServer/InstallableApps/awb.war
```

2 Deploy the resultant `awb` folder to the new managed server.

Note: Deploying the expanded Web archive enables editing of runtime descriptor elements without requiring redeployment.

3 During deployment, provide:

- Select **Install this deployment as an application**
- Select the managed server you created as the application deployment target; for example, `EPMAWeb`
- Select **I will make the deployment accessible from the following location**, and ensure the pre-filled file system location for the expanded archive is correct
- Start servicing all requests for the deployed application, switching **State** from **Prepared** to **Active**

4 Perform the steps in "Configuring Web Server Routing" on page 91.

Deploying Performance Management Architect DataSync

Configuring Managed Server

Note: This document assumes your managed server is named `DataSync`.

➤ To configure managed server:

1 Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".

2 Add the following Java argument to the Server Start configuration for the managed server:

`-Dhyperion.home=`*`HYPERION_HOME`*

3 Add a condition to `startEPMSsystem.cmd|sh`. See "Evaluating SERVER_NAME in Start Script" on page 57.

4 If necessary, add a library search path variable. See "Library Search Path Variables" on page 18.

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- If SAP is defined in Shared Services as a user directory, add

  `HYPERION_HOME``/common/SAP/bin`

Windows example:

```
:DataSync
set PATH=%PATH%;%HYPERION_HOME%\common\SAP\bin
set PATH=%HYPERION_HOME%\common\CSS\9.5.0.0\bin
goto STARTWLS
```

UNIX and Linux example:

```
DataSync )
  LD_LIBRARY_PATH=${HYPERION_HOME}/common/SAP/bin:${LD_LIBRARY_PATH}
  export LD_LIBRARY_PATH
  ;;
```

5 **Restart the managed server.**

Deploying DataSync.war

➤ To deploy `DataSync.war`:

1 **Create a script to expand the following Web archive to a file system accessible to the WebLogic domain:**

HYPERION_HOME`/products/Foundation/BPMA/AppServer/InstallableApps/DataSync.war`

Windows example:

```
set PATH=C:\bea\jdk150_10\bin;%PATH%
mkdir C:\Hyperion\deployments\WebLogic9\servers\BPMA\webapps\DataSync
cd /d C:\Hyperion\deployments\WebLogic9\servers\BPMA\webapps\DataSync
jar -xf C:\Hyperion\products\Foundation\BPMA\AppServer\InstallableApps\DataSync.war
pause
```

UNIX and Linux example:

```
PATH=/opt/bea/jdk150_10/bin:${PATH} ; export PATH
mkdir -p /opt/hyperion/deployments/WebLogic9/servers/BPMA/webapps/DataSync
cd /opt/hyperion/deployments/WebLogic9/servers/BPMA/webapps/awb/DataSync
jar -xf /opt/hyperion/products/Foundation/BPMA/AppServer/InstallableApps/DataSync.war
```

2 **Deploy the resultant `DataSync` folder to the newly created managed server.**

Note: Deploying the expanded Web archive enables editing of runtime descriptor elements without requiring redeployment.

3 **During deployment, provide:**

- Select **Install this deployment as an application**
- Select the managed server you created as the application deployment target; for example, `DataSync`
- Select **I will make the deployment accessible from the following location**, and ensure the pre-filled file system location for the expanded archive is correct
- Start servicing all requests for the deployed application, switching **State** from **Prepared** to **Active**

4 **Perform the steps in "Configuring Web Server Routing" on page 91.**

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open:**

`http://HOSTNAME.EXAMPLE.COM:LISTEN_PORT/awb/`

2 **Review the output for your WebLogic managed server in** `DOMAIN_HOME/servers/SERVER_NAME/logs`.

3 **Review the product logs in** `HYPERION_HOME/logs/epma`.

Deploying Calculation Manager

Configuring Managed Server

Note: This document assumes your managed server is named `Calcmgr`.

➤ To configure managed server:

1 **Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".**

2 **Add the following Java argument to the Server Start configuration for the managed server:**

`-Dhyperion.home=HYPERION_HOME`

3 **Add a condition to** `startEPMSsystem.cmd`. **See "Evaluating SERVER_NAME in Start Script" on page 57.**

4 **If necessary, add a library search path variable. See "Library Search Path Variables" on page 18.**

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- If SAP is defined in Shared Services as a user directory, add

  `HYPERION_HOME/common/SAP/bin`

Windows example:

```
:Calcmgr
set PATH=%PATH%;%HYPERION_HOME%\common\SAP\bin
set PATH=%HYPERION_HOME%\common\CSS\9.5.0.0\bin
goto STARTWLS
```

UNIX and Linux example:

```
Calcmgr )
  LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:${HYPERION_HOME}/common/SAP/bin
  export LD_LIBRARY_PATH
  ;;
```

5 **Restart the managed server.**

Deploying calcmgr.ear

➤ To deploy `calcmgr.ear`:

1 **Create a script to expand the following Enterprise archive to a file system accessible to your WebLogic domain:**

HYPERION_HOME`/products/Foundation/CALC/AppServer/InstallableApps/calcmgr.ear`

Windows script example:

```
set PATH=C:\bea\jdk150_10\bin;%PATH%
mkdir C:\Hyperion\deployments\WebLogic9\servers\calcmgr\webapps\calcmgr
cd /d C:\Hyperion\deployments\WebLogic9\servers\calcmgr\webapps\
jar -xf C:\Hyperion\products\Foundation\CALC\AppServer\InstallableApps\calcmgr.ear
calcmgr.war
cd calcmgr
jar -xf ..\calcmgr.war
del ..\calcmgr.war
pause
```

UNIX and Linux

```
#!/bin/sh

PATH=/opt/bea/jdk150_10/bin:${PATH} ; export PATH
mkdir -p /opt/hyperion/deployments/WebLogic9/servers/calcmgr/webapps/calcmgr
cd /opt/hyperion/deployments/WebLogic9/servers/calcmgr/webapps/calcmgr
jar -xf /opt/hyperion/products/Foundation/CALC/AppServer/InstallableApps/calcmgr.ear
calcmgr.war
jar -xf calcmgr.war
rm calcmgr.war
```

2 **Deploy the resultant `calcmgr` folder to the newly created managed server.**

Note: Deploying the expanded Web archive enables editing of runtime descriptor elements without requiring redeployment.

3 **During deployment, provide:**

- Select **Install this deployment as an application**
- Select the managed server you created as the application deployment target; for example, `Calcmgr`
- Select **I will make the deployment accessible from the following location**, and ensure the pre-filled file system location for the expanded archive is correct

- Start servicing all requests for the deployed application, switching **State** from **Prepared** to **Active**

4 Perform the steps in "Configuring Web Server Routing" on page 91.

Verifying Deployment

➤ To verify deployment:

1 Using a Web browser, open:

`http://HOSTNAME.EXAMPLE.COM:LISTEN_PORT/calcmgr/`

2 Review the output for your WebLogic managed server in *DOMAIN_HOME*`/servers/`*SERVER_NAME*`/logs`.

3 Review the product logs in *HYPERION_HOME*`/logs/calcmgr`.

Deploying Financial Reporting

Configuring Managed Server

Note: This document assumes your managed server is named `FinancialReporting`.

➤ To configure managed server:

1 Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".

2 Add the Java arguments to the Server Start configuration for the managed server:

- `-DHYPERION_HOME=`*HYPERION_HOME*
- `-Xss296k`

3 UNIX only:

Execute:

HYPERION_HOME`/products/biplus/InstallableApps/freporting_web.env`.

Copy the output, and set the library search path variable in the script used to start WebLogic, for details see step 4.

4 Add the following:

a. Add a condition to `startEPMSsystem.cmd|sh`. See "Evaluating SERVER_NAME in Start Script" on page 57.

b. **UNIX:** Add an environment variable DISPLAY, and assign a valid X11 address as its value. See "Enabling an X Virtual Frame Buffer for Financial Reporting and Web Analysis" in the Oracle Enterprise Performance Management System Installation and Configuration Guide.

Windows example:

```
:FinancialReporting
goto STARTWLS
```

UNIX example:

```
FinancialReporting )
LD_LIBRARY_PATH=/opt/hyperion/products/biplus/bin:/opt/hyperion/products/biplus/
bin64:/opt/hyperion/common/EssbaseRTC/9.5.0.0/bin:/opt/hyperion/common/ADM/9.5.0.0/
Essbase/9.5.0.0/bin:/opt/hyperion/common/ADM/Planning/9.5.0.0/bin:/opt/hyperion/
common/SAP/bin:$LD_LIBRARY_PATH
    export LD_LIBRARY_PATH

    DISPLAY=xvfb-host.example.com:99.0
    export DISPLAY
    ;;
```

5 **Restart the managed server.**

Deploying HReports.ear

➤ To deploy `HReports.ear`:

1 **Create a script to expand the following Enterprise archive to a file system accessible to your WebLogic domain:**

HYPERION_HOME/products/biplus/InstallableApps/HReports.ear

Windows script example:

```
set PATH=C:\bea\jdk150_10\bin;%PATH%
mkdir C:\Hyperion\deployments\WebLogic9\servers\FinancialReporting\webapps\hr
cd /d C:\Hyperion\deployments\WebLogic9\servers\FinancialReporting\webapps
jar -xf C:\Hyperion\products\biplus\InstallableApps\HReports.ear hr.war
cd hr
jar -xf ..\hr.war
del ..\hr.war
pause
```

UNIX script example:

```
PATH=/opt/bea/jdk150_10/bin:${PATH} ; export PATH
mkdir -p /opt/hyperion/deployments/WebLogic9/servers/FinancialReporting/webapps/hr
cd /opt/hyperion/deployments/WebLogic9/servers/FinancialReporting/webapps
jar -xf /opt/hyperion/products/biplus/InstallableApps/HReports.ear hr.war
cd hr && jar -xf ../hr.war
rm ../hr.war
```

2 **Deploy the resultant `hr` folder to the newly created managed server.**

Note: Deploying the expanded Web archive enables editing of runtime descriptor elements without requiring redeployment.

3 **During deployment, provide:**

- Select **Install this deployment as an application**
- Select the managed server you created as the application deployment target; for example, `FinancialReporting`
- Select **I will make the deployment accessible from the following location**, and ensure the pre-filled file system location for the expanded archive is correct
- Start servicing all requests for the deployed application, switching **State** from **Prepared** to **Active**

4 **Perform the steps in** **"Configuring Web Server Routing" on page 91**.

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open**

`http://`*`HOSTNAME.EXAMPLE.COM:LISTEN_PORT`*`/hr/status.jsp`

2 **Review the output for your WebLogic managed server in:**

`DOMAIN_HOME``/servers/`*`SERVER_NAME`*`/logs`

3 **Review the product logs in:**

`HYPERION_HOME``/logs/BIPlus`

Deploying Web Analysis

Configuring Managed Server

Note: This document assumes your managed server is named `WebAnalysis`.

➤ To configure managed server:

1 **Complete the steps in** **Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps"**.

2 **Add the following Java arguments to the Server Start configuration for the managed server:**

```
-DHYPERION_HOME=HYPERION_HOME
-DBIPLUS_HOME=HYPERION_HOME/products/biplus
```

```
-DESS_ES_HOME=HYPERION_HOME/products/biplus/appsinfo/WebAnalysis/EssbaseJAPI
-Datf.session.binding.kill-with-prejudice=true
```

3 **Add the following CLASSPATH entries to the Server Start configuration for the managed server:**

```
HYPERION_HOME/common/CLS/9.5.0.0/lib/cls-9_5_0.jar
HYPERION_HOME/common/JDBC/DataDirect/3.7/lib/hyjdbc.jar
HYPERION_HOME/common/SAP/lib
```

Note: Separate entries with semi-colons (;) for Windows, colons (:) for UNIX.

4 **Add the following:**

a. Add a condition to

```
DOMAIN_HOME/bin/startEPMSystem.cmd|sh
```

that defines following environment variables for the managed server:

- BIPLUS_HOME
- ESSLANG
- ARBORPATH
- ESS_ES_HOME
- ICU_DATA

See "Evaluating SERVER_NAME in Start Script" on page 57.

b. Add a library search path variable. See "Library Search Path Variables" on page 18.

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- If SAP is defined in Shared Services as a user directory, add

  ```
  HYPERION_HOME/common/SAP/bin
  ```

c. UNIX: Add an environment variable DISPLAY, and assign a valid X11 address as its value. See "Enabling an X Virtual Frame Buffer for Financial Reporting and Web Analysis" in the *Oracle Enterprise Performance Management System Installation and Configuration Guide*.

Windows example:

```
:WebAnalysis
set BIPLUS_HOME=%HYPERION_HOME%\products\biplus
set ESSLANG=English_UnitedStates.Latin1@Binary
set ARBORPATH=%HYPERION_HOME%\common\EssbaseRTC\9.5.0.0
set ESSBASEPATH=%ARBORPATH%
set ESS_ES_HOME=%BIPLUS_HOME%\bin\EssbaseJAPI
set ICU_DATA=%HYPERION_HOME%\common\ADM\9.5.0.0\Essbase\9.5.0.0\bin\HssEssDriver
set PATH=%BIPLUS_HOME%/bin
set PATH=%PATH%;%ARBORPATH%/bin
set PATH=%PATH%;%HYPERION_HOME%/common/ADM/9.5.0.0/Essbase/9.5.0.0/bin
set PATH=%PATH%;%HYPERION_HOME%/products/FinancialManagement/Client
set PATH=%PATH%;%HYPERION_HOME%/products/FinancialManagement/common
```

```
set PATH=%PATH%;%HYPERION_HOME%/common/SAP/bin
set PATH=%PATH%;%HYPERION_HOME%/common/CSS/9.5.0.0/bin
goto STARTWLS
```

UNIX and Linux example:

```
WebAnalysis )
  DISPLAY=xvfb-host.example.com:99.0
  export DISPLAY
  BIPLUS_HOME=${HYPERION_HOME}/products/biplus
  export BIPLUS_HOME
  ESSLANG=English_UnitedStates.Latin1@Binary
  export ESSLANG
  ARBORPATH=${HYPERION_HOME}/common/EssbaseRTC/9.5.0.0
  export ARBORPATH
  ESSBASEPATH=${ARBORPATH}
  export ESSBASEPATH
  ESS_ES_HOME=${BIPLUS_HOME}/bin/EssbaseJAPI
  export ESS_ES_HOME
  ICU_DATA=${HYPERION_HOME}/common/ADM/9.5.0.0/Essbase/9.5.0.0/bin/HssEssDriver
  export ICU_DATA
  LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:${BIPLUS_HOME}/bin
  LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:${ARBORPATH}/bin
  LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:${HYPERION_HOME}/common/ADM/9.5.0.0/Essbase/9.5.
0.0/bin
  LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:${HYPERION_HOME}/common/SAP/bin
  export LD_LIBRARY_PATH
  ;;
```

5 **Restart the managed server.**

Deploying WebAnalysis.ear

➤ To deploy `WebAnalysis.ear`:

1 **Create a script to expand the following Enterprise archive to a file system accessible to your WebLogic domain:**

HYPERION_HOME`/products/biplus/InstallableApps/WebAnalysis.ear`

Windows script example:

```
set PATH=C:\bea\jdk150_10\bin;%PATH%
mkdir C:\Hyperion\deployments\WebLogic9\servers\WebAnalysis\webapps\WebAnalysis
cd /d C:\Hyperion\deployments\WebLogic9\servers\WebAnalysis\webapps
jar -xf C:\Hyperion\products\biplus\InstallableApps\WebAnalysis.ear WebAnalysis.war
cd WebAnalysis C:> jar -xf ..\WebAnalysis.war
del ..\WebAnalysis.war
pause
```

UNIX and Linux script example:

```
PATH=/opt/bea/jdk150_10/bin:${PATH} ; export PATH
mkdir -p /opt/hyperion/deployments/WebLogic9/servers/WebAnalysis/webapps/WebAnalysis
cd /opt/hyperion/deployments/WebLogic9/servers/WebAnalysis/webapps
jar -xf /opt/hyperion/products/biplus/InstallableApps/WebAnalysis.ear
WebAnalysis.war
```

```
cd WebAnalysis && jar -xf ../WebAnalysis.war
rm ../WebAnalysis.war
```

2 **Deploy the resultant `WebAnalysis` folder to the newly created managed server.**

Note: Deploying the expanded Web archive enables editing of runtime descriptor elements without requiring redeployment.

3 **During deployment, provide:**

- Select **Install this deployment as an application**
- Select the managed server you created as the application deployment target; for example, `WebAnalysis`
- Select **I will make the deployment accessible from the following location**, and ensure the pre-filled file system location for the expanded archive is correct
- Start servicing all requests for the deployed application, switching **State** from **Prepared** to **Active**

4 **Perform the steps in "Configuring Web Server Routing" on page 91.**

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open**

`http://`*`HOSTNAME.EXAMPLE.COM:LISTEN_PORT`*`/WebAnalysis/`

2 **Review the output for your WebLogic managed server in** *`DOMAIN_HOME`*`/servers/`*`SERVER_NAME`*`/logs`.

3 **Review the product logs in** *`HYPERION_HOME`*`/logs/biplus`.

Deploying Planning

Configuring Managed Server

Note: This document assumes your managed server is named `HyperionPlanning`.

➤ To configure managed server:

1 **Start the RMIRegistry for Planning using:**

- **Windows:** Windows service
- **UNIX and Linux:** Execute: *HYPERION_HOME*/common/RMI/9.5.0.0/HyperionRMIRegistry

2 **Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".**

3 **Add the following Java argument to the Server Start configuration for the managed server:**

-DHYPERION_HOME=*HYPERION_HOME*

4 **If deploying to WebLogic powered by a 64-bit JVM:**

a. On Solaris or AIX, additionally add the Java argument: -Dcom.hyperion.planning.datamodel=64

b. On Solaris, additionally add the Java argument: -d64

5 **Add the following:**

a. Add a condition to

DOMAIN_HOME/bin/startEPMSystem.cmd|sh

that defines PLANNING_HOME as the environment variable for the managed server.

See "Evaluating SERVER_NAME in Start Script" on page 57.

b. Add a library search path variable. See "Library Search Path Variables" on page 18.

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- If SAP is defined in Shared Services as a user directory, add

  HYPERION_HOME/common/SAP/bin
- If deploying to WebLogic powered by a 64-bit JVM on Solaris or AIX, include EssbaseRTC-64 and Planning/lib64 in the library search path.

Windows example:

```
:HyperionPlanning
set PLANNING_HOME=%HYPERION_HOME%/products/Planning
set PATH=%PATH%;%HYPERION_HOME%/common/EssbaseRTC/9.5.0.0/bin
set PATH=%PATH%;%HYPERION_HOME%/common/SAP/bin
goto STARTWLS
```

UNIX and Linux example:

```
HyperionPlanning )
  PLANNING_HOME=${HYPERION_HOME}/products/Planning
  export PLANNING_HOME
  LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:${HYPERION_HOME}/common/EssbaseRTC/9.5.0.0/bin
  LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:${HYPERION_HOME}/common/SAP/bin
  LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:${HYPERION_HOME}/products/Planning/lib
  export LD_LIBRARY_PATH

LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:${HYPERION_HOME}/common/EssbaseRTC/9.5.0.0/bin

JAVA_OPTIONS="-server -DComponentName=HyperionPlanning -DHYPERION_HOME=
<HYPERION_HOME > -Dhyperion.home=<HYPERION_HOME>
```

```
-Dweblogic.j2ee.application.tmpDir=<tmp_location> ${JAVA_OPTIONS} "
export JAVA_OPTIONS
```

6 **Change the Cookie name in the Weblogic.xml file to:**

```
<session-param>
      <param-name>CookieName</param-name>
      <param-value>HPSESSIONID</param-value>
```

7 **Restart the managed server.**

Deploying HyperionPlanning.ear

➤ To deploy `HyperionPlanning.ear`:

1 **Create a script to expand the following Enterprise archive to a file system accessible to the WebLogic domain:**

HYPERION_HOME`/products/Planning/AppServer/InstallableApps/common/HyperionPlanning.ear`

Windows script example:

```
set PATH=C:\bea\jdk150_10\bin;%PATH%
mkdir C:\Hyperion\deployments\WebLogic9\servers\HyperionPlanning\webapps
\HyperionPlanning
cd /d C:\Hyperion\deployments\WebLogic9\servers\HyperionPlanning\webapps
jar -xf C:\Hyperion\products\Planning\AppServer\InstallableApps\Common
\HyperionPlanning.ear HyperionPlanning.war
cd HyperionPlanning C:> jar -xf ..\HyperionPlanning.war
del ..\HyperionPlanning.war
pause
```

UNIX and Linux script example:

```
PATH=/opt/bea/jdk150_10/bin:${PATH} ; export PATH
mkdir -p /opt/hyperion/deployments/WebLogic9/servers/HyperionPlanning/webapps/
HyperionPlanning
cd /opt/hyperion/deployments/WebLogic9/servers/HyperionPlanning/webapps
jar -xf /opt/hyperion/products/Planning/AppServer/InstallableApps/common/
HyperionPlanning.ear HyperionPlanning.war
cd HyperionPlanning && jar -xf ../HyperionPlanning.war
rm ../HyperionPlanning.war
```

2 **Deploy the expanded `HyperionPlanning` folder to the newly created managed server.**

Note: Deploying the expanded Web archive enables editing of runtime descriptor elements without requiring redeployment.

3 **During deployment, provide:**

- Select **Install this deployment as an application**
- Select the managed server you created as the application deployment target; for example, `HyperionPlanning`
- Select **I will make the deployment accessible from the following location**, and ensure the pre-filled file system location for the expanded archive is correct

- Start servicing all requests for the deployed application, switching **State** from **Prepared** to **Active**

4 Perform the steps in "Configuring Web Server Routing" on page 91.

Verifying Deployment

➤ To verify deployment:

1 Using a Web browser, open:

`http://HOSTNAME.EXAMPLE.COM:LISTEN_PORT/HyperionPlanning/LogOn.jsp`

2 Review the output for your WebLogic managed server in *DOMAIN_HOME*`/servers/`*SERVER_NAME*`/logs`.

3 Review the product logs in *HYPERION_HOME*`/logs/Planning`.

Deploying Performance Scorecard

Configuring Managed Server

Note: This document assumes your managed server is named `Scorecard`.

➤ To configure managed server:

1 Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".

2 Add the following Java arguments to the Server Start configuration for the managed server:

- `-Dhyperion.home=`*HYPERION_HOME*
- `-DESS_ES_HOME=`*HYPERION_HOME*`/common/EssbaseRTC/9.5.0.0`

3 Add the following:

a. Add a condition to

DOMAIN_HOME`/bin/startEPMSystem.cmd|sh`

that defines following environment variables for the managed server:

- ARBORPATH
- ESSBASEPATH
- ESSLANG

See "Evaluating SERVER_NAME in Start Script" on page 57.

b. Add a library search path variable. See "Library Search Path Variables" on page 18.

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- If SAP is defined in Shared Services as a user directory, add

  HYPERION_HOME/common/SAP/bin

Windows example:

```
:Scorecard
set ARBORPATH=%HYPERION_HOME%\common\EssbaseRTC\9.5.0.0
set ESSBASEPATH=%ARBORPATH%
set ESSLANG=English_UnitedStates.Latin1@Binary
set PATH=%ARBORPATH%\bin
set PATH=%PATH%;%HYPERION_HOME%\common\ADM\9.5.0.0\Essbase\9.5.0.0\bin
set PATH=%PATH%;%HYPERION_HOME%\common\SAP\bin
set PATH=%PATH%;%HYPERION_HOME%\common\CSS\9.5.0.0\bin
goto STARTWLS
```

UNIX and Linux example:

```
Scorecard )
HYPERION_HOME=/opt/hyperion
export HYPERION_HOME
  ARBORPATH=${HYPERION_HOME}/common/EssbaseRTC/9.5.0.0
  export ARBORPATH
  ESSBASEPATH=${ARBORPATH}
  export ESSBASEPATH
  ESSLANG=English_UnitedStates.Latin1@Binary
  export ESSLANG
  LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:${ARBORPATH}/bin
  LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:${HYPERION_HOME}/common/ADM/9.5.0.0/Essbase/9.5.
0.0/bin
  LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:${HYPERION_HOME}/common/SAP/bin
  export LD_LIBRARY_PATH
  ;;
```

4 **Restart the managed server.**

Deploying HPSWebReports.war and HPSAlerter.war

➤ To deploy `HPSWebReports.war` and `HPSAlerter.war`:

1 **Create a script to expand following Web archives to a file system accessible to the WebLogic domain:**

HYPERION_HOME/products/PerformanceScorecard/AppServer/InstallableApps/common/webapps/config/HPSWebReports.war
HYPERION_HOME/products/PerformanceScorecard/AppServer/InstallableApps/common/webapps/config/HPSAlerter.war

Windows script example:

```
set PATH=C:\bea\jdk150_10\bin;%PATH%
mkdir C:\Hyperion\products\PerformanceScorecard\AppServer\InstallableApps\common
\webapps\config\HPSWebReports
mkdir C:\Hyperion\products\PerformanceScorecard\AppServer\InstallableApps\common
\webapps\config\HPSAlerter
cd C:\Hyperion\products\PerformanceScorecard\AppServer\InstallableApps\common\webapps
\config\HPSWebReports
jar -xf C:\Hyperion\products\PerformanceScorecard\AppServer\InstallableApps\common
\webapps\config\HPSWebReports.war
cd C:\Hyperion\products\PerformanceScorecard\AppServer\InstallableApps\common\webapps
\config\Alerter
jar -xf C:\Hyperion\products\PerformanceScorecard\AppServer\InstallableApps\common
\webapps\config\HPSAlerter.war
pause
```

UNIX and Linux script example:

```
#!/bin/sh
PATH=/opt/bea/jdk150_10/bin:${PATH}; export PATH
mkdir -p /opt/hyperion/deployments/WebLogic9/servers/Scorecard/webapps/HPSWebReports
cd /opt/hyperion/deployments/WebLogic9/servers/Scorecard/webapps/HPSWebReports
jar -xf /opt/hyperion/products/PerformanceScorecard/AppServer/InstallableApps/common/
webapps/configured/HPSWebReports.war
mkdir -p /opt/hyperion/deployments/WebLogic9/servers/Scorecard/webapps/HPSAlerter
cd /opt/hyperion/deployments/WebLogic9/servers/Scorecard/webapps/HPSAlerter
jar -xf /opt/hyperion/products/PerformanceScorecard/AppServer/InstallableApps/common/
webapps/configured/HPSAlerter.war
```

2 **Deploy the resultant `HPSWebReports` and `HPSAlerter` folders to the new managed server.**

Note: Deploying the expanded Web archive enables editing of runtime descriptor elements without requiring redeployment.

3 **During deployment, provide:**

- Select **Install this deployment as an application**
- Select the managed server you created as the application deployment target; for example, `Scorecard`
- Select **I will make the deployment accessible from the following location**, and ensure the pre-filled file system location for the expanded archive is correct
- Start servicing all requests for the deployed application, switching **State** from **Prepared** to **Active**

4 **Perform the steps in "Configuring Web Server Routing" on page 91.**

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open:**

`http://`*`HOSTNAME.EXAMPLE.COM:LISTEN_PORT`*`/workspace/`

2 Access the Oracle Hyperion Performance Scorecard, Fusion Edition module using the EPM Workspace navigation menu.

3 Review the output for your WebLogic managed server in *DOMAIN_HOME*/servers/*SERVER_NAME*/logs.

4 Review the product logs in *HYPERION_HOME*/logs/hps.

Deploying Profitability and Cost Management

Configuring Managed Server

Note: This document assumes your managed server is named Profitability.

➤ To configure managed server:

1 Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".

2 Add the following Java arguments to the Server Start configuration for the managed server:

- -DHYPERION_HOME=*HYPERION_HOME*
- -Dhyperion.home=*HYPERION_HOME*

3 Add a condition to startEPMSsystem.cmd|sh. See "Evaluating SERVER_NAME in Start Script" on page 57.

4 If necessary, add a library search path variable. See "Library Search Path Variables" on page 18.

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- If SAP is defined in Shared Services as a user directory, add *HYPERION_HOME*/common/SAP/bin

Windows example:

```
:Profitability
set PATH=%PATH%;%HYPERION_HOME%\common\SAP\bin
set PATH=%HYPERION_HOME%\common\CSS\9.5.0.0\bin
goto STARTWLS
```

UNIX and Linux example:

```
Profitability )
  LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:${HYPERION_HOME}/common/SAP/bin
```

```
export LD_LIBRARY_PATH
;;
```

5 Restart the managed server.

Deploying profitability.war

➤ To deploy `profitability.war`:

1 Create a script to expand the following Web archive to a file system accessible to your WebLogic domain:

HYPERION_HOME/products/Profitability/AppServer/InstallableApps/common/profitability.war

Windows script example:

```
set PATH=C:\bea\jdk150_10\bin;%PATH%
mkdir C:\Hyperion\deployments\WebLogic9\servers\Profitability\webapps\profitability
cd /d C:\Hyperion\deployments\WebLogic9\servers\Profitability\webapps\profitability
jar -xf C:\Hyperion\products\Profitability\AppServer\InstallableApps\common
\profitability.war
pause
```

UNIX and Linux script example:

```
#!/bin/sh
PATH=/opt/bea/jdk150_10/bin:${PATH} ; export PATH
mkdir -p /opt/hyperion/deployments/WebLogic9/servers/Profitability/webapps/
profitability
cd /opt/hyperion/deployments/WebLogic9/servers/Profitability/webapps/profitability
jar –xf /opt/hyperion/products/Profitability/AppServer/InstallableApps/common/
profitability.war
```

2 Deploy the resultant `Profitability` folder to the newly created managed server.

Note: Deploying the expanded Web archive enables editing of runtime descriptor elements without requiring redeployment.

3 During each deployment, provide:

- Select **Install this deployment as an application**
- Select the managed server you created as the application deployment target; for example, `Profitability`
- Select **I will make the deployment accessible from the following location**, and ensure the pre-filled file system location for the expanded archive is correct
- Start servicing all requests for the deployed application, switching **State** from **Prepared** to **Active**

4 Perform the steps in "Configuring Web Server Routing" on page 91.

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open:**

`http://HOSTNAME.EXAMPLE.COM:LISTEN_PORT/Profitability/ping.jsp`

2 **Review the output for your WebLogic managed server in** *DOMAIN_HOME*`/servers/`*SERVER_NAME*`/logs`.

3 **Review the product logs in** *HYPERION_HOME*`/logs/profitability`.

Deploying ERP Integrator

- "Creating Data Sources in WebLogic" on page 88
- "Configuring Managed Server" on page 80
- "Deploying aif.war" on page 90
- "Verifying Deployment" on page 83

Creating Data Sources in WebLogic

If you are using WebLogic with Oracle Hyperion Financial Data Quality Management ERP Integration Adapter for Oracle Applications, you need to manually create the data source for your ERP Integrator application before deploying.

➤ To create a data source for ERP Integrator using WebLogic:

1 **From the WebLogic Administrative Console, select Services, then JDBC, and then Data sources.**

2 **Click Lock and Edit, then New. Enter the data source configuration details:**

   a. Enter the JNDI name by which `aif.war` will refer to this data source (One way to create this link is adding `weblogic.xml` with resource-ref-name set to "`jdbc/aifDS`" and jndi-name set to the JNDI name specified here; for example: jdbc/aifDS - as done by configtool when configuring automatically, to `aif.war` before deployment.) The JNDI name for the data source must be `jdbc/aifDS`.

   b. Enter values for other options; for example: drivers, host name, user, password. The database this data source refers to should be the one used for Oracle Hyperion Financial Data Quality Management ERP Integration Adapter for Oracle Applications database configuration task. Clear "Supports Global Transactions". Set target server for this data source to the server where `aif.war` is, or will be, deployed.

3 **Click Finish, then click Activate changes.**

Configuring Managed Server

Note: This document assumes your managed server is named `aif`.

➤ To configure managed server:

1 **Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".**

2 **Add the following Java argument to the Server Start configuration for the managed server:**

`-DHYPERION_HOME=`*`HYPERION_HOME`*

3 **If deploying to WebLogic powered by a 64-bit JVM:**

a. On Solaris or AIX, additionally add the Java argument: `-Dcom.hyperion.aif.datamodel=64`

b. On Solaris, additionally add the Java argument: `-d64`

4 **Add the following:**

a. Add a condition to

`DOMAIN_HOME``/bin/startEPMSystem.cmd|sh`

that defines AIF_HOME as the environment variable for the managed server.

See "Evaluating SERVER_NAME in Start Script" on page 57.

b. Add a library search path variable. See "Library Search Path Variables" on page 18.

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- If SAP is defined in Shared Services as a user directory, add

  `HYPERION_HOME``/common/SAP/bin`

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- If SAP is defined in Shared Services as a user directory, add

  `HYPERION_HOME``/common/SAP/bin`

- If deploying to WebLogic powered by a 64-bit JVM on Solaris or AIX, include EssbaseRTC-64 and `aif/lib64` in the library search path.

Windows example:

```
:AIF
set AIF_HOME=%HYPERION_HOME%/products/FinancialDataQuality
set PATH=%PATH%;%HYPERION_HOME%/common/EssbaseRTC/9.5.0.0/bin
set PATH=%PATH%;%HYPERION_HOME%/common/SAP/bin
goto STARTWLS
```

UNIX and Linux example:

```
AIF )
  AIF_HOME=${HYPERION_HOME}/products/FinancialDataQuality
  export AIF_HOME
  LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:${HYPERION_HOME}/common/EssbaseRTC/9.5.0.0/bin
  LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:${HYPERION_HOME}/common/SAP/bin
  LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:${HYPERION_HOME}/products/FinancialDataQuality/
lib
  export LD_LIBRARY_PATH

LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:${HYPERION_HOME}/common/EssbaseRTC/9.5.0.0/bin

JAVA_OPTIONS="-server -DComponentName=AIF -DHYPERION_HOME= <HYPERION_HOME >
-Dhyperion.home=<HYPERION_HOME> -Dweblogic.j2ee.application.tmpDir=<tmp_location> $
{JAVA_OPTIONS} "
export JAVA_OPTIONS
```

5 **Restart the managed server.**

Deploying aif.war

➤ To deploy `aif.war`:

1 **Create a script to expand the following Enterprise archive to a file system accessible to the WebLogic domain:**

HYPERION_HOME/products/FinancialDataQuality/AppServer/InstallableApps/aif.war

The target directory when deployed by config is:

HYPERION_HOME/deployments/WebLogic9/servers/ERPIntegrator

Windows script example:

```
set PATH=C:\bea\jdk150_10\bin;%PATH%
mkdir C:\Hyperion\deployments\WebLogic9\servers\ERPIntegrator\webapps\aif
cd /d C:\Hyperion\deployments\WebLogic9\servers\ERPIntegrator\webapps\aif
jar -xf C:\Hyperion\products\FinancialDataQuality\AppServer\InstallableApps\aif.war
pause
```

UNIX and Linux script example:

```
PATH=/opt/bea/jdk150_10/bin:${PATH} ; export PATH
mkdir -p /opt/hyperion/deployments/WebLogic9/servers/ERPIntegrator/webapps/aif
cd /opt/hyperion/deployments/WebLogic9/servers/ERPIntegrator/webapps/aif
jar -xf /opt/hyperion/products/FinancialDataQuality/AppServer/InstallableApps/
aif.war
```

2 **Deploy the expanded `AIF` folder to the newly created managed server.**

Note: Deploying the expanded Web archive enables editing of runtime descriptor elements without requiring redeployment.

3 **During deployment, provide:**

- Select **Install this deployment as an application**

- Select the managed server you created as the application deployment target; for example, `AIF`
- Select **I will make the deployment accessible from the following location**, and ensure the pre-filled file system location for the expanded archive is correct
- Start servicing all requests for the deployed application, switching **State** from **Prepared** to **Active**

4 Perform the steps in "Configuring Web Server Routing" on page 91.

Verifying Deployment

➤ To verify deployment:

1 Using a Web browser, open:

`http://`*`HOSTNAME.EXAMPLE.COM:LISTEN_PORT`*`/aif/faces/setup/Main.jsp`

2 Review the output for your WebLogic managed server in *`DOMAIN_HOME`*`/servers/`*`SERVER_NAME`*`/logs`.

3 Review the product logs in *`HYPERION_HOME`*`/logs/aif`.

Configuring Web Server Routing

This section describes how to configure the WebLogic Web server Apache and IIS plug-ins.

Apache HTTP Server

Follow the instructions in the BEA WebLogic *Using Web Server Plug-ins with WebLogic Server* guide to enable the WebLogic Web server Apache plug-in. For example, add the following to your Apache server configuration file:

```
LoadModule weblogic_module WL_HOME/server/plugin/OS/ARCH/mod_wl_20.so
```

Tip: Apache HTTP Server is included with the EPM System, and is installed in *`HYPERION_HOME`*`/common/httpServers/Apache`.

After you enable the plug-in, add the following directives to your Apache server configuration file:

```
<IfModule mod_weblogic.c>
    WLForwardUriUnparsed ON
    KeepAliveEnabled ON
    KeepAliveSecs 20
</IfModule>
```

Shared Services

Add the following directives to your Apache server configuration file, replacing *WL_SERVER* and *WL_PORT* with your actual WebLogic managed server hostname and port.

```
RedirectMatch 301 ^/interop$ /interop/
<LocationMatch /interop/*>
       SetHandler weblogic-handler
       PathTrim /
       WeblogicCluster WL_SERVER:WL_PORT
</LocationMatch>
```

EPM Workspace

Add the following directives to your Apache server configuration file, replacing:

- *WL_SERVER* and *WL_PORT* with your actual WebLogic managed server hostname and port
- *HYPERION_HOME* with your actual location

If you have clustered EPM Workspace across more than one managed server, separate each *WL_SERVER*:*WL_PORT* entry with a comma; for example, *WL_SERVER1*:*WL_PORT1*,*WL_SERVER2*:*WL_PORT2*.

```
RedirectMatch 301 ^/workspace$ /workspace/
<LocationMatch /workspace/(DynamicHelp|search|PDFView|servlet|portletservlet|
administration|browse|personalpages)>
    SetHandler weblogic-handler
    PathTrim /
    WeblogicCluster WL_SERVER:WL_PORT
</LocationMatch>
<LocationMatch /workspace/(viewmanager|jobmanager|ihtml|dataaccess|logon|js|modules|
media|conf|wsrp4j)>
    SetHandler weblogic-handler
    PathTrim /
    WeblogicCluster WL_SERVER:WL_PORT
</LocationMatch>
<LocationMatch /workspace/(cdsrpc|logon|prefs|BPMContext|ResourceProxy|)$>
    SetHandler weblogic-handler
    PathTrim /
    WeblogicCluster WL_SERVER:WL_PORT
</LocationMatch>
<LocationMatch /workspace/(.*)\.(jsp|jsv|jsw)$>
    SetHandler weblogic-handler
    PathTrim /
    WeblogicCluster WL_SERVER:WL_PORT
</LocationMatch>
<LocationMatch /biplus_webservices/*>
SetHandler weblogic-handler
PathTrim /
WeblogicCluster WL_SERVER:WL_PORT
</LocationMatch>
AliasMatch /workspace/(resources|css|docs|images|img|SmartView|themes|thirdparty|
wsmedia|zeroadmin|CrystalBall)/(.*) HYPERION_HOME/products/Foundation/workspace/
AppServer/InstalledApps/workspace_static/$1/$2
```

```
Alias /wsmedia "HYPERION_HOME/products/Foundation/workspace/AppServer/InstalledApps/
workspace_static/wsmedia"
Alias /InsightInstaller "HYPERION_HOME/products/Foundation/workspace/AppServer/
InstalledApps/workspace_static/zeroadmin/component/Insight"
```

Performance Management Architect

Add the following directives to your Apache server configuration file, replacing *WL_SERVER* and *WL_PORT*with your actual WebLogic managed server hostname and port.

If you have clustered Oracle Hyperion EPM Architect, Fusion Edition across more than one managed server, separate each *WL_SERVER*:*WL_PORT* entry with a comma; for example, *WL_SERVER1*:*WL_PORT1*,*WL_SERVER2*:*WL_PORT2*.

```
RedirectMatch 301 ^/awb$ /awb/
RedirectMatch 301 ^/DataSync$ /DataSync/
<LocationMatch /awb/*>
    SetHandler weblogic-handler
    PathTrim /
    WeblogicCluster WL_SERVER:WL_PORT
</LocationMatch>
<LocationMatch /DataSync/*>
    SetHandler weblogic-handler
    PathTrim /
    WeblogicCluster WL_SERVER:WL_PORT

</LocationMatch>
```

Calculation Manager

Add the following directives to your Apache server configuration file, replacing *WL_SERVER* and *WL_PORT* with your actual WebLogic managed server hostname and port.

If you have clustered Hyperion Calculation Manager across more than one managed server, separate each *WL_SERVER*:*WL_PORT* entry with a comma; for example, *WL_SERVER1*:*WL_PORT1*,*WL_SERVER2*:*WL_PORT2*.

```
RedirectMatch 301 ^/calcmgr$ /calcmgr/
<LocationMatch /calcmgr/*>
    SetHandler weblogic-handler
    PathTrim /
    WeblogicCluster WL_SERVER:WL_PORT
</LocationMatch>
```

Administration Services

Add the following directives to your Apache server configuration file, replacing *WL_SERVER* and *WL_PORT* with your actual WebLogic managed server hostname and port.

```
RedirectMatch 301 ^/eas$ /eas/
RedirectMatch 301 ^/easconsole$ /easconsole/
RedirectMatch 301 ^/easdocs$ /easdocs/
RedirectMatch 301 ^/hbrlauncher$ /hbrlauncher/
<LocationMatch ^/eas/*>
```

```
    SetHandler weblogic-handler
    PathTrim /
    WeblogicCluster WL_SERVER:WL_PORT
</LocationMatch>
<LocationMatch /easconsole/*>
    SetHandler weblogic-handler
    PathTrim /
    WeblogicCluster WL_SERVER:WL_PORT
</LocationMatch>
<LocationMatch /easdocs/*>
    SetHandler weblogic-handler
    PathTrim /
    WeblogicCluster WL_SERVER:WL_PORT
</LocationMatch>
```

Provider Services

Add the following directives to your Apache server configuration file, replacing *WL_SERVER* and *WL_PORT* with your actual WebLogic managed server hostname and port.

If you have clustered Provider Services across more than one managed server, separate each *WL_SERVER*:*WL_PORT* entry with a comma; for example, *WL_SERVER1*:*WL_PORT1*,*WL_SERVER2*:*WL_PORT2*.

```
RedirectMatch 301 ^/aps$ /aps/
<LocationMatch /aps/*>
    SetHandler weblogic-handler
    PathTrim /
    WeblogicCluster WL_SERVER:WL_PORT
</LocationMatch>
```

Application Builder for .NET

Add the following directives to your Apache server configuration file, replacing *WL_SERVER* and *WL_PORT* with your actual WebLogic managed server hostname and port.

If you have clustered Oracle's Hyperion® Application Builder for .NET across more than one managed server, separate each *WL_SERVER*:*WL_PORT* entry with a comma; for example, *WL_SERVER1*:*WL_PORT1*,*WL_SERVER2*:*WL_PORT2*.

```
RedirectMatch 301 ^/EssbaseObjects$ /EssbaseObjects/
<LocationMatch /EssbaseObjects/*>
    SetHandler weblogic-handler
    PathTrim /
    WeblogicCluster WL_SERVER:WL_PORT
</LocationMatch>
```

Financial Reporting

Add the following directives to your Apache server configuration file, replacing *WL_SERVER* and *WL_PORT* with your actual WebLogic managed server hostname and port.

If you have clustered Oracle Hyperion Financial Reporting, Fusion Edition across more than one managed server, separate each *WL_SERVER*:*WL_PORT* entry with a comma; for example, *WL_SERVER1*:*WL_PORT1*,*WL_SERVER2*:*WL_PORT2*.

```
RedirectMatch 301 ^/hr$ /hr/
<LocationMatch /hr/*>
    SetHandler weblogic-handler
    PathTrim /
    WeblogicCluster WL_SERVER:WL_PORT
</LocationMatch>
```

Web Analysis

Add the following directives to your Apache server configuration file, replacing *WL_SERVER* and *WL_PORT*with your actual WebLogic managed server hostname and port.

If you have clustered Web Analysis across more than one managed server, separate each *WL_SERVER*:*WL_PORT* entry with a comma; for example, *WL_SERVER1*:*WL_PORT1*,*WL_SERVER2*:*WL_PORT2*.

```
RedirectMatch 301 ^/WebAnalysis$ /WebAnalysis/
<LocationMatch /WebAnalysis/(ServerConsole|templates|portletservlet|portlettemplates|
servlet|modules|resources|wa_javadocs)>
    SetHandler weblogic-handler
    PathTrim /
    WeblogicCluster WL_SERVER:WL_PORT
</LocationMatch>
<LocationMatch /WebAnalysis/(processor|DirectoryServlet|Config|config|)$>
    SetHandler weblogic-handler
    PathTrim /
    WeblogicCluster WL_SERVER:WL_PORT
</LocationMatch>
<LocationMatch /WebAnalysis/(.*)\.(jsp|xml|jar|exe)$>
    SetHandler weblogic-handler
    PathTrim /
    WeblogicCluster WL_SERVER:WL_PORT
</LocationMatch>
Alias /WebAnalysis "HYPERION_HOME/products/Foundation/workspace/AppServer/InstalledApps/
WebAnalysis_static"
```

Planning

Add the following directives to your Apache server configuration file, replacing *WL_SERVER* and *WL_PORT* with your actual WebLogic managed server hostname and port.

If you have clustered Planning across more than one managed server, you must:

- Set the `WLCookieName` cookie session ID to *HPSESSIONID*.
- List each *WL_SERVER*:*WL_PORT* separately and separate each *WL_SERVER*:*WL_PORT* entry with a comma; for example, *WL_SERVER1*:*WL_PORT1*,*WL_SERVER2*:*WL_PORT2*.

```
<LocationMatch /HyperionPlanning/*>
    SetHandler weblogic-handler
    athTrim /
    KeepAliveEnabled ON
```

```
    KeepAliveSecs 20
    WLCookieName HPSESSIONID
    WeblogicCluster WL_SERVER:WL_PORT
</LocationMatch>
```

Performance Scorecard

Add the following directives to your Apache server configuration file, replacing *WL_SERVER* and *WL_PORT*with your actual WebLogic managed server hostname and port.

If you have clustered Oracle Hyperion EPM Architect, Fusion Edition across more than one managed server, separate each *WL_SERVER*:*WL_PORT* entry with a comma; for example, *WL_SERVER1*:*WL_PORT1*,*WL_SERVER2*:*WL_PORT2*.

```
RedirectMatch 301 ^/HPSWebReports$ /HPSWebReports/
RedirectMatch 301 ^/HPSAlerter$ /HPSAlerter/
<LocationMatch /HPSWebReports/*>
    SetHandler weblogic-handler
    PathTrim /
    WeblogicCluster WL_SERVER:WL_PORT
</LocationMatch>
<LocationMatch /HPSAlerter/*>
    SetHandler weblogic-handler
    PathTrim /
    WeblogicCluster WL_SERVER:WL_PORT
</LocationMatch>
```

Profitability and Cost Management

Add the following directives to your Apache server configuration file, replacing *WL_SERVER* and *WL_PORT* with your actual WebLogic managed server hostname and port.

If you have clustered Oracle Hyperion Profitability and Cost Management, Fusion Edition across more than one managed server, separate each *WL_SERVER*:*WL_PORT* entry with a comma; for example, *WL_SERVER1*:*WL_PORT1*,*WL_SERVER2*:*WL_PORT2*.

```
RedirectMatch 301 ^/profitability$ /profitability/
<LocationMatch /profitability/*>
    SetHandler weblogic-handler
    PathTrim /
    WeblogicCluster WL_SERVER:WL_PORT
</LocationMatch>
```

ERP Integrator

Add the following directives to your Apache server configuration file, replacing *WL_SERVER* and *WL_PORT* with your actual WebLogic managed server hostname and port.

```
RedirectMatch 301 ^/aif$ /aif/
<LocationMatch /aif/*>
    SetHandler weblogic-handler
    PathTrim /
    WeblogicCluster WL_SERVER:WL_PORT
</LocationMatch>
```

Oracle BI EE

Add the following directives to your Apache server configuration file, replacing *WL_SERVER* and *WL_PORT* with your actual WebLogic managed server hostname and port.

```
RedirectMatch 301 ^/analytics$ /analytics
<LocationMatch /analytics/*>
    SetHandler weblogic-handler
    PathTrim /
    WeblogicCluster WL_SERVER:WL_PORT
</LocationMatch>

RedirectMatch 301 ^/xmlpserver$ /xmlpserver/
    <LocationMatch /xmlpserver /*>
    SetHandler weblogic-handler
    PathTrim /
WeblogicCluster WL_SERVER:WL_PORT
</LocationMatch>
```

IIS

The steps that follow are for example only and may require alteration for your configuration. Refer to the *BEA WebLogic Using Web Server Plug-ins with Oracle WebLogic Server* guide for complete instructions.

Copying Plug-in Binaries

Each EPM System Web application that uses IIS to proxy requests must have its own copy of the following:

- WebLogic IIS proxy plug-in binary (`iisproxy.dll`)
- Configuration file (`iisproxy.ini`)

One copy of the following are shared by all EPM System Web applications:

- WebLogic IIS forward plug-in binary (`iisforward.dll`)
- Configuration file (`iisproxy.ini`)

For example:

```
mkdir C:\Hyperion\WL-VHOSTS\SharedServices
mkdir C:\Hyperion\WL-VHOSTS\Workspace
copy C:\bea\weblogic92\server\plugin\win\ARCH\iisforward.dll C:\Hyperion\WL-VHOSTS
copy C:\bea\weblogic92\server\plugin\win\ARCH\iisproxy.dll C:\Hyperion\WL-VHOSTS
\SharedServices
copy C:\bea\weblogic92\server\plugin\win\ARCH\iisproxy.dll C:\Hyperion\WL-VHOSTS
\Workspace
```

Creating the Shared Copy of iisproxy.ini

In the same directory where the IIS forward plug-in binary (`iisforward.dll`) was copied, create `iisproxy.ini`. This file will be used by IIS to proxy requests from EPM Workspace and Web Analysis Web applications. Create two entries:

```
WlForwardPath=/
PathTrim=/
```

The first line will be updated during the configuration for each EPM System Web application that includes static content.

Registering iisforward.dll as an IIS filter

➤ To configure proxying by path:

1 **Register `iisforward.dll` as an ISAPI filter for your Web Site.**

Windows example:

Name: `WLForwardFilter`

Path: `C:\Hyperion\WL-VHOSTS\iisforward.dll`

2 **If using IIS 6.0, or later: Allow `iisforward.dll` in Web server Extensions.**

Creating Private Copies of iisproxy.ini

➤ To create private copies of `iisproxy.ini`:

1 **For each directory where the proxy plug-in binary (`iisproxy.dll`) was copied, create the file `iisproxy.ini`.**

2 **Add the following parameters to the file you create, replacing *WL_SERVER* and *WL_PORT* with your actual WebLogic managed server hostname and port. If you have clustered across more than one managed server, separate each *WL_SERVER*:*WL_PORT* entry with a comma; for example, *WL_SERVER1*:*WL_PORT1*, *WL_SERVER2*:*WL_PORT2*.**

Note: Parameters are case-sensitive.

```
KeepAliveEnabled=true
KeepAliveSecs=20
WebLogicCluster=WL_SERVER:WL_PORT
```

3 **If using IIS 6.0, or later: Allow each copy of `iisproxy.dll` in the Web server Extensions.**

Shared Services

➤ To configure IIS for use with Shared Services:

1 **If using IIS 6.0, or later, create an application pool; for example, `SharedServicesAppPool`.**

2 Create a new **Virtual Directory** with the attributes:

- **Alias**: interop (case-sensitive)
- **Path**: *IISPROXY_DLL_DIR*; for example, *HYPERION_HOME*/WL-VHOSTS/SharedServices
- Permissions: Read, Run scripts, Execute

3 Do one:

- IIS 5.0: In the properties for the `interop` **Virtual Directory**, set **Application Protection** to High (Isolated)
- IIS 6.0, or later: In the properties for the `interop` **Virtual Directory**, select the **Application Pool** you created; for example, `SharedServicesAppPool`

4 Using the full path to the copy of `iisproxy.dll` that resides in the file system directory you created for the **Virtual Directory** called `interop`, **Insert a Wildcard application map**.

5 If using IIS 6.0, or later, deselect **Verify file exists**.

EPM Workspace

Note: You need 2 virtual directories, but only one application pool, one web service extension, and one copy of the iisproxy.ini files.

➤ To configure IIS for use with EPM Workspace:

1 Using the dynamic content forward paths identified inAppendix A, "Forward Paths For Web Server Routing" as reference, add the following parameters to the `WlForwardPath` line of iisproxy.ini file from the same directory where `iisforward.dll` is located; for example,

```
C:\Hyperion\WL-VHOSTS\iisproxy.ini

/workspace/DynamicHelp,/workspace/search,/workspace/PDFView,/workspace/servlet,/
workspace/portletservlet,/workspace/cdsrpc,/workspace/administration,/workspace/
browse,/workspace/personalpages,/workspace/viewmanager,/workspace/jobmanager,/
workspace/ihtml,/workspace/dataaccess,/workspace/logon,/workspace/prefs,/workspace/
js,/workspace/modules,/workspace/media,/workspace/conf,/workspace/BPMContext,/
workspace/wsrp4j,/workspace/ResourceProxy,/workspace/*.jsp,/workspace/*.jsv,/
workspace/*.jsw
```

2 If using IIS 6.0, or later, create an application pool; for example, `WorkspaceAppPool`

3 Create a **Virtual Directory** with the attributes:

- **Alias**: workspace (case-sensitive
- **Path**: *HYPERION_HOME*/products/Foundation/workspace/AppServer/InstalledApps/workspace_static
- **Permissions**: Read, Run scripts, Execute

4 Create a **Virtual Directory** with the attributes:

- **Alias**: wsmedia (case-sensitive)

- **Path**: *HYPERION_HOME*/products/Foundation/workspace/AppServer/InstalledApps/workspace_static/wsmedia
- **Permissions**: Read, Run scripts, Execute

5 Create a **Virtual Directory** with the attributes:

- **Alias**: InsightInstaller (case-sensitive)
- **Path**: *HYPERION_HOME*/products/Foundation/workspace/AppServer/InstalledApps/workspace_static/zeroadmin/component/Insight
- **Permissions**: Read, Run scripts, Execute

6 Do one:

- IIS 5.0: In the properties for the workspace **Virtual Directory**, set **Application Protection** to High (Isolated)
- IIS 6.0, or later: In the properties for the workspace **Virtual Directory**, select the **Application Pool** you created; for example, WorkspaceAppPool

7 Add .wlforward as an application extension for the **Executable** iisproxy.dll to the **Virtual Directory** called workspace.

8 If using IIS 6.0, deselect **Verify file exists**.

Performance Management Architect

Note: You need 2 virtual directories, but only one application pool, one web service extension, and one copy of the iisproxy.ini files.

➤ To configure IIS for use with Performance Management Architect:

1 If using IIS 6.0, or later, create one application pool; for example, BPMAAppPool

2 Create a **Virtual Directory** with the attributes:

- **Alias**: awb (case-sensitive)
- **Path**: *IISPROXY_DLL_DIR*; for example, *HYPERION_HOME*/WL-VHOSTS/BPMA
- **Permissions**: Read, Run scripts, Execute

3 Create a **2nd Virtual Directory** with the attributes:

- **Alias**: DataSync (case-sensitive)
- **Path**: *IISPROXY_DLL_DIR*; for example, *HYPERION_HOME*/WL-VHOSTS/BPMA
- **Permissions**: Read, Run scripts, Execute

4 Do one:

- IIS 5.0: In the properties for the awb **Virtual Directory**, set **Application Protection** to High (Isolated).
- IIS 6.0, or later: In the properties for the awb **Virtual Directory**, select the **Application Pool** you created; for example, BPMAppPool

5 **Using the full path to the copy of `iisproxy.dll` that resides in the file system directory you created for the Virtual Directory called `awb`, Insert a Wildcard application map.**

6 **If using IIS 6.0, deselect Verify file exists.**

Calculation Manager

➤ To configure IIS for use with Calculation Manager:

1 **If using IIS 6.0, or later, create an application pool; for example, `CalcAppPool`**

2 **Create a Virtual Directory with the attributes:**

- **Alias**: `calcmgr` (case-sensitive)
- **Path**: *`IISPROXY_DLL_DIR`*; for example, *`HYPERION_HOME`*`/WL-VHOSTS/calcmgr`
- **Permissions**: Read, Run scripts, Execute

3 **Do one:**

- IIS 5.0: In the properties for the `calcmgr` **Virtual Directory**, set **Application Protection** to High (Isolated).
- IIS 6.0, or later: In the properties for the calcmgr **Virtual Directory**, select the **Application Pool** you created; for example, `CalcAppPool`

4 **Using the full path to the copy of `iisproxy.dll` that resides in the file system directory you created for the Virtual Directory `calcmgr`, Insert a Wildcard application map.**

5 **If using IIS 6.0, deselect Verify file exists.**

Administration Services

➤ To configure IIS for use with Administration Services

1 **If using IIS 6.0, or later, create an application pool; for example, `EASAppPool`**

2 **Create a Virtual Directory with the attributes:**

- **Alias**: eas (case-sensitive)
- **Path** : *IISPROXY_DLL_DIR*; for example, *`HYPERION_HOME`*`/WL-VHOSTS/EAS`
- **Permissions**: Read, Run scripts, Execute

3 **Do one:**

- For IIS 5.0: In the properties for the `eas` **Virtual Directory**, set **Application Protection** to High (Isolated)
- for IIS 6.0, or later: In the properties for the `eas` **Virtual Directory**, select the **Application Pool** you created; for example, `EASAppPool`.

4 **Using the full path to the copy of `iisproxy.dll` that resides in the file system directory you created for the Virtual Directory called `eas`, Insert a Wildcard application map.**

5 **If using IIS 6.0, deselect Verify file exists.**

6 Repeat these steps to create a **Virtual Directory** for each of these aliases: `easconole`, `easdocs`, `hbrlauncher`.

Provider Services

➤ To configure IIS for use with Provider Services:

1 If using IIS 6.0, or later, create an application pool; for example, `APSAppPool`

2 Create a **Virtual Directory** with the attributes:

- **Alias**: aps (case-sensitive)
- **Path**: *`IISPROXY_DLL_DIR`*`; for example, `*`HYPERION_HOME`*`/WL-VHOSTS/APS`
- **Permissions**: Read, Run scripts, Execute

3 Do one:

- IIS 5.0: In the properties for the `eas` **Virtual Directory**, set **Application Protection** to High (Isolated)
- IIS 6.0, or later: In the properties for the `eas` **Virtual Directory**, select the **Application Pool** you created; for example, `APSAppPool`

4 Using the full path to the copy of `iisproxy.dll` that resides in the file system directory you created for the **Virtual Directory** called `aps`, **Insert** a **Wildcard application map**.

5 If using IIS 6.0, deselect **Verify file exists**.

Application Builder for .NET

➤ To configure IIS for use with Application Builder for .NET:

1 If using IIS 6.0, or later, create an application pool; for example, `HABAppPool`.

2 Create a **Virtual Directory** with the attributes:

- **Alias**: EssbaseObjects (case-sensitive)
- **Path**: *`IISPROXY_DLL_DIR`*; for example, *`HYPERION_HOME`*`/WL-VHOSTS/HAB`
- **Permissions**: Read, Run scripts, Execute

3 Do one:

- IIS 5.0: In the properties for the `EssbaseObjects` **Virtual Directory**, set **Application Protection** to High (Isolated).
- IIS 6.0, or later: In the properties for the `EssbaseObjects` **Virtual Directory**, select the **Application Pool** you created; for example, `HABAppPool`.

4 Using the full path to the copy of `iisproxy.dll` that resides in the file system directory you created for the **Virtual Directory** `EssbaseObjects`, **Insert** a **Wildcard application map**.

5 If using IIS 6.0, deselect **Verify file exists**.

Financial Reporting

➤ To configure IIS for use with Financial Reporting:

1 **If using IIS 6.0, or later, create an application pool; for example,** `FRAppPool`

2 **Create a Virtual Directory with the attributes:**

- **Alias**: hr (case-sensitive)
- **Path**: `IISPROXY_DLL_DIR`; for example,

  HYPERION_HOME`/WL-VHOSTS/FinancialReporting`
- **Permissions**: Read, Run scripts, Execute

3 **Do one:**

- IIS 5.0: In the properties for the `hr` **Virtual Directory**, set **Application Protection** to High (Isolated)
- IIS 6.0, or later: In the properties for the `hr` **Virtual Directory**, select the **Application Pool** you created; for example, `FRAppPool`

4 **Using the full path to the copy of** `iisproxy.dll` **that resides in the file system directory you created for the Virtual Directory called** `hr`**, Insert a Wildcard application map.**

5 **If using IIS 6.0, deselect Verify file exists.**

Web Analysis

➤ To configure IIS for use with Web Analysis:

1 **Using the dynamic content forward paths identified in**Appendix A, "Forward Paths For Web Server Routing" **as reference, add the following parameters to the** `WlForwardPath` **line of** `iisproxy.ini` **file from same directory where** `iisforward.dll` **is located; for example,**

```
C:\Hyperion\WL-VHOSTS\iisproxy.ini

/WebAnalysis/ServerConsole,/WebAnalysis/templates,/WebAnalysis/hfmtemplates,/
WebAnalysis/hitemplates,/WebAnalysis/portlettemplates,/WebAnalysis/servlet,/
WebAnalysis/modules,/WebAnalysis/resources,/WebAnalysis/processor,/WebAnalysis/
DirectoryServlet,/WebAnalysis/config,/WebAnalysis/Config,/WebAnalysis/wa_javadocs/
*,/WebAnalysis/*.jsp,/WebAnalysis/*.xml,/WebAnalysis/*.jar,/WebAnalysis/*.exe
```

2 **If using IIS 6.0, or later, create an application pool; for example,** `WAAppPool`

3 **Create a Virtual Directory with the attributes:**

- **Alias**: WebAnalysis (case-sensitive)
- **Path**: *HYPERION_HOME*`/products/Foundation/workspace/AppServer/InstalledApps/WebAnalysis_static`
- **Permissions**: Read, Run scripts, Execute

4 **Navigate to Properties of WebAnalysis virtual Directory and add** `index_en.html` **(or** `index_YOUR_LOCALE.html`**) in default content page list on Documents tab.**

5 **Do one:**

- IIS 5.0: In the properties for the `WebAnalysis` **Virtual Directory**, set **Application Protection** to High (Isolated)
- IIS 6.0, or later: In the properties for the `WebAnalysis` **Virtual Directory**, select the **Application Pool** you created; for example, `WAAppPool`

6 **Add `.wlforward` as an application extension for the Executable `iisproxy.dll` to the Virtual Directory called `WebAnalysis`.**

7 **If using IIS 6.0, deselect Verify file exists.**

Planning

➤ To configure IIS for use with Planning:

1 **If using IIS 6.0, or later, create an application pool; for example, `HPAppPool`**

2 **Create a Virtual Directory with the attributes:**

- **Alias**: `HyperionPlanning` (case-sensitive)
- **Path**: *IISPROXY_DLL_DIR*; for example, *`HYPERION_HOME`*`/WL-VHOSTS/Planning`
- **Permissions**: Read, Run scripts, Execute

3 **Do one:**

- IIS 5.0: In the properties for the `HyperionPlanning` **Virtual Directory**, set **Application Protection** to High (Isolated)
- IIS 6.0, or later: In the properties for the `HyperionPlanning` **Virtual Directory**, select the **Application Pool** you created; for example, `HPAppPool`

4 **Using the full path to the copy of `iisproxy.dll` that resides in the file system directory you created for the Virtual Directory called `HyperionPlanning`, Insert a Wildcard application map.**

5 **If using IIS 6.0, deselect Verify file exists.**

Performance Scorecard

Note: 2 virtual directories are required, but only one application pool, one web service extension, and one copy of the iisproxy.ini files.

➤ To configure IIS for use with Performance Scorecard:

1 **If using IIS 6.0, or later, create an application pool; for example, `HPSAppPool`**

2 **Create a Virtual Directory with the attributes:**

- **Alias**: `HPSWebReports` (case-sensitive)
- **Path**: *IISPROXY_DLL_DIR*; for example, *`HYPERION_HOME`*`/WL-VHOSTS/HPS`
- **Permissions**: Read, Run scripts, Execute

3 **Create a 2nd Virtual Directory with the attributes:**

- **Alias**: `HPSAlerter` (case-sensitive)
- **Path**: *IISPROXY_DLL_DIR*; for example, *HYPERION_HOME*`/WL-VHOSTS/HPS`
- **Permissions**: Read, Run scripts, Execute

4 **Do one:**

- IIS 5.0: In the properties for the `HPSWebReports` **Virtual Directory**, set **Application Protection** to High (Isolated).
- IIS 6.0, or later: In the properties for the `HPSWebReports` **Virtual Directory**, select the **Application Pool** you created; for example, `HPSAppPool`

5 **Using the full path to the copy of `iisproxy.dll` that resides in the file system directory you created for the Virtual Directory called `HPSWebReports`, Insert a Wildcard application map.**

6 **If using IIS 6.0, deselect Verify file exists.**

Profitability and Cost Management

➤ To configure IIS for use with Profitability and Cost Management:

1 **If using IIS 6.0, or later, create an application pool; for example, `ProfitabailityAppPool`**

2 **Create a Virtual Directory with the attributes:**

- **Alias**: profitability (case-sensitive)
- **Path**: *IISPROXY_DLL_DIR*; for example, *HYPERION_HOME*`/WL-VHOSTS/Profitability`
- **Permissions**: Read, Run scripts, Execute

3 **Do one:**

- IIS 5.0: In the properties for the profitability **Virtual Directory**, set **Application Protection** to High (Isolated).
- IIS 6.0, or later: In the properties for the profitability **Virtual Directory**, select the **Application Pool** you created; for example, profitability

4 **Using the full path to the copy of `iisproxy.dll` that resides in the file system directory you created for the Virtual Directory `profitability`, Insert a Wildcard application map.**

5 **If using IIS 6.0, deselect Verify file exists.**

ERP Integrator

➤ To configure IIS for use with ERP Integrator:

1 **If using IIS 6.0, or later, create an application pool; for example, AIFAppPool.**

2 **Create a Virtual Directory with the attributes:**

- **Alias:** aif (case-sensitive)
- **Path:** `IISPROXY_DLL_DIR`, for example,

HYPERION_HOME/WL-VHOSTS/ ERPIntegrator

- **Permissions:** Read, Run scripts, Execute

3 **Do one:**

- IIS 5.0: In the properties for the `aif` **Virtual Directory**, set **Application Protection** to High (Isolated)
- IIS 6.0, or later: In the properties for the `aif` **Virtual Directory**, select the **Application Pool** you created; for example, `AIFAppPool`

4 **Using the full path to the copy of `iisproxy.dll` that resides in the file system directory you created for the Virtual Directory called `aif`, Insert a Wildcard application map.**

5 **If using IIS 6.0, or later, deselect Verify file exists.**

Oracle BI EE

➤ To configure IIS for use with Oracle BI EE:

1 **If using IIS 6.0, or later, create an application pool; for example, `AnalyticsAppPool` and `PublisherAppPool`.**

2 **Create a Virtual Directory with the attributes:**

- **Alias:** analytics, xmlpserver (case-sensitive)
- **Path:** *IISPROXY_DLL_DIR*; for example,

  HYPERION_HOME/WL-VHOSTS/Analytics, *HYPERION_HOME*/WL-VHOSTS/Publisher
- **Permissions:** Read, Run scripts, Execute

3 **Do one:**

- IIS 5.0: In the properties for the **Virtual Directory**, set **Application Protection** to High (Isolated).
- IIS 6.0, or later: In the properties for the **Virtual Directory**, select the **Application Pool** you created; for example, `AnalyticsAppPool`.

4 **Using the full path to the copies of `iisproxy.dll` that resides in the file system directory you created for the Virtual Directories `analytics` and `xmlpserver`, Insert a Wildcard application map.**

5 **If using IIS 6.0, deselect Verify file exists.**

Configuring Financial Management with a Web Server

To configure Financial Management with Apache Web Server: using a text editor, open the Web server's configuration file (`httpd.conf`), and add the following directives, replacing *HFM_HOST* and *HFM_PORT* with real values:

```
 #Change modules/mod_proxy.so as needed; e.g. libexec/mod_proxy on UNIX
<IfModule !mod_proxy.c>
LoadModule proxy_module modules/mod_proxy.so
</IfModule>
```

```
<IfModule !mod_proxy_connect.c>
LoadModule proxy_connect_module modules/mod_proxy_connect.so
</IfModule>
<IfModule !mod_proxy_http.c>
LoadModule proxy_http_module modules/mod_proxy_http.so
</IfModule>
#ProxyRequests Off
ProxyPreserveHost On
ProxyPass /hfm http://HFM_HOST:HFM_PORT/hfm
ProxyPassReverse /hfm http://HFM_HOST:HFM_PORT/hfm
```

where *HFM_HOST* is the Financial Management host machine and *HFM_PORT* is the default port, 80.

Using Only IIS with Financial Management

For using only IIS (for example, if Apache is not used at all): You must configure the IIS used by Financial Management as the front-end Web server for EPM Workspace. You can have the EPM Workspace Web application server running on a separate machine, but you must install a copy of the EPM Workspace component onto the machine running IIS. The EPM Workspace installation includes files that are served by the Web server, such as the help files.

After installing EPM Workspace on the IIS machine, use Life Cycle Management to change the Web server hostname.

5 Deploying EPM System into WebSphere Application Server

In This Chapter

WebSphere Environment Assumptions

This guide assumes that the following are defined in the WebSphere administration console, using the specified page:

- **Java arguments:** `Application servers > SERVER_NAME > Java and Process Management > Process Definition > Java Virtual Machine > Generic JVM arguments`
- **CLASSPATH:** `Application servers > SERVER_NAME > Java and Process Management > Process Definition > Java Virtual Machine > Classpath`
- **Environment Variables:** `Application servers > SERVER_NAME > Java and Process Management > Process Definition > Environment Entries`

Note: **Environment Entries** is called **Custom Properties** in older releases of WebSphere. Do not confuse with `Application servers > SERVER_NAME > Process Definition > Java Virtual Machine > Custom Properties`.

WebSphere Prerequisites

Install a WebSphere supported Web server, and WebSphere Web server plug-ins.

To configure, see "Configuring Web Server Routing" on page 135.

Tip: Apache HTTP Server is included with the EPM System, and is installed in *HYPERION_HOME*/common/httpServers/Apache.

Deploying Shared Services

Configuring the Web Application Server Instance

➤ To configure the Web application server instance:

1 **Start OpenLDAP using the Windows service, or by running**

HYPERION_HOME/products/Foundation/openLDAP/startService.bat|startOpenLDAP.sh

2 **Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".**

3 **Add the following Java arguments:**

-DHYPERION_HOME=*HYPERION_HOME*
-Dhyperion.home=*HYPERION_HOME*

See "WebSphere Environment Assumptions" on page 109.

4 **Add the following CLASSPATH entries:**

HYPERION_HOME/deployments/WebSphere6/SharedServices9/config
HYPERION_HOME/common/JDBC/DataDirect/3.7/lib/hyjdbc.jar
HYPERION_HOME/common/JakartaCommons/commons-dbcp-1.2.1.jar
HYPERION_HOME/common/JakartaCommons/commons-pool-1.3.jar
HYPERION_HOME/common/SAP/lib

Note: Separate entries with semi-colons (;) for Windows, colons (:) for UNIX.

See "WebSphere Environment Assumptions" on page 109.

5 **Add the following:**

a. Add an environment variable: HYPERION_HOME.

b. If necessary, add a library search path variable. See "Library Search Path Variables" on page 18.

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- If SAP is defined in Shared Services as a user directory, add

  HYPERION_HOME/common/SAP/bin

Windows example:

```
HYPERION_HOME   C:\Hyperion
PATH C:\Hyperion\common\CSS\9.5.0.0\bin;C:\Hyperion\common\SAP\bin
```

UNIX and Linux example:

```
HYPERION_HOME   /opt/hyperion
LD_LIBRARY_PATH   /opt/hyperion/common/SAP/bin
```

Deploying interop.war

➤ To deploy `interop.war`:

1 **Deploy the following Web archive to the new Web application server instance:**

HYPERION_HOME/deployments/WebSphere6/SharedServices9/config/interop.war

2 **During deployment, provide:**

- **Context root:** `/interop` (case-sensitive; do not alter.)
- Enable **Precompile JavaServer Pages files**
- **Optional:** Provide a descriptive **Application Name**: for example, `SharedServices`
- **Map modules to servers**: Select the correct Web application server and Web server, if applicable; for example `SharedServices` and `webserver1`

3 **Perform the steps in** **"Configuring Web Server Routing" on page 135**.

Applying Post Deployment Application Settings

➤ To apply post deployment application settings:

1 **Enable Session Override for the deployed application,**

then change **Cookie path** to `/interop`

2 **Change the Class loader order for the deployed application to Classes loaded with application class loader first.**

Note: In older releases of WebSphere the administration console property to be edited was called **Class Loading and File Update Detection**, and the value to select was called **Parent Last**.

3 **Restart your Web application server instance.**

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open:**

`http://HOSTNAME.EXAMPLE.COM:LISTEN_PORT/interop/`

2 **Log on to Shared Services.**

3 **Review the output for your WebSphere Web application server instance in** `WAS_ROOT/profiles/PROFILE_NAME/logs`.

4 **Review the product logs in** `HYPERION_HOME/logs/SharedServices9`.

Deploying Administration Services

Configuring the Web Application Server Instance

➤ To configure the Web application server instance:

1 **Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".**

2 **Add the following Java arguments to the Server Start configuration for the managed server:**

```
-Dhyperion.home=HYPERION_HOME
-DHYPERION_HOME=/HYPERION_HOME
-DESS_ES_HOME=/HYPERION_HOME/products/Essbase/eas/server
-DEAS_HOME=HYPERION_HOME/products/Essbase/eas
-DEAS_LOG_LEVEL=5000
-DEAS_LOG_LOCATION=HYPERION_HOME/logs/eas/easserver.log
-DEAS_SERVER_VERSION=11.1.1.4
```

See "WebSphere Environment Assumptions" on page 109.

3 **Add the CLASSPATH entry:**

`HYPERION_HOME/common/JakartaCommons/commons-lang-2.1.jar`

See "WebSphere Environment Assumptions" on page 109.

4 **If you are deploying onto WebSphere 6.1, select Process Definition > Environment Entries.**

5 **Add the following:**

a. Add the following environment variables to the configuration for the Web application server instance:

- HYPERION_HOME

- EAS_HOME
- ESSLANG
- ARBORPATH
- ESSBASEPATH

b. Add a library search path variable. See "Library Search Path Variables" on page 18.

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- If SAP is defined in Shared Services as a user directory, add

  HYPERION_HOME/common/SAP/bin

Windows example:

```
HYPERION_HOME     C:\Hyperion
EAS_HOME          C:\Hyperion\products\Essbase\eas
ESSLANG           English_UnitedStates.Latin1@Binary
ARBORPATH         C:\Hyperion\products\Essbase\eas\server
ESSBASEPATH       C:\Hyperion\products\Essbase\eas\server

PATH  C:\Hyperion\products\Essbase\eas\server\bin;C:\Hyperion\common\CSS\9.5.0.
0\bin;C:\Hyperion\common\SAP\bin
```

UNIX and Linux example:

```
HYPERION_HOME     /opt/hyperion
EAS_HOME          /opt/hyperion/products/Essbase/eas
ESSLANG           English_UnitedStates.Latin1@Binary
ARBORPATH         /opt/hyperion/products/Essbase/eas/server
ESSBASEPATH       /opt/hyperion/products/Essbase/eas/server

LD_LIBRARY_PATH   /opt/hyperion/products/Essbase/eas/server/bin:/opt/hyperion/common/
SAP/bin
```

Deploying eas.ear

➤ To deploy `eas.ear`:

1 **Deploy the following enterprise archive to the Web application server instance that you created:**

HYPERION_HOME/products/Essbase/eas/server/AppServer/InstallableApps/common/eas.ear

2 **During deployment, provide:**

- Enable **Precompile JavaServer Pages files**
- **Optional:** Provide a descriptive **Application Name**: for example, `EAS`
- **Map modules to servers**: Select the correct Web application server and Web server, if applicable; for example `EAS` and `webserver1`

3 **Perform the steps in "Configuring Web Server Routing" on page 135.**

Applying Post Deployment Application Settings

➤ To apply post deployment application settings:

1 **Enable Session Override for the deployed application,**

then change **Cookie path** to `/eas`.

2 **Restart the Web application server instance.**

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open:**

`http://`*`HOSTNAME.EXAMPLE.COM:LISTEN_PORT`*`/easconsole/console.html`

2 **Log on to Administration Services using the Java Web Start console.**

3 **Review the output for the WebSphere Web application server instance in** *`WAS_ROOT`*`/profiles/`*`PROFILE_NAME`*`/logs`.

4 **Review the product logs in** *`HYPERION_HOME`*`/logs/eas`.

Deploying Provider Services

Configuring the Web Application Server Instance

➤ To configure the Web application server instance:

1 **Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".**

2 **Add the Java arguments:**

```
–Dhyperion.home=HYPERION_HOME
-DESS_ES_HOME=HYPERION_HOME/products/Essbase/aps
```

See "WebSphere Environment Assumptions" on page 109.

3 **Add the following:**

a. Add the environment variables:

- HYPERION_HOME
- ESSBASEPATH

b. Add a library search path variable. See "Library Search Path Variables" on page 18.

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- If SAP is defined in Shared Services as a user directory, add

  HYPERION_HOME/common/SAP/bin

Windows example:

```
HYPERION_HOME C:\Hyperion
ESSBASEPATH   C:\Hyperion\products\Essbase\aps
PATH...C:\Hyperion\products\Essbase\aps\bin;C:\Hyperion\common\CSS\9.5.0.0\bin;C:
\Hyperion\common\SAP\bin
```

UNIX and Linux example:

```
HYPERION_HOME /opt/hyperion
ESSBASEPATH /opt/hyperion/products/Essbase/aps
LD_LIBRARY_PATH /opt/hyperion/products/Essbase/eas/server/bin:/opt/
hyperion/common/SAP/bin
```

Deploying aps.war

➤ To deploy `aps.war`:

1 **Deploy the following Web archive to the Web application server instance that you created:**

HYPERION_HOME/products/Essbase/aps/AppServer/InstallableApps/common/aps.war

2 **During deployment, provide**

- **Context root:** `/aps` (Case-sensitive; do not alter.)
- Enable **Precompile JavaServer Pages files**
- **Optional:** Provide a descriptive **Application Name**: for example, `APS`.
- **Map modules to servers**: Select the correct Web application server and Web server, if applicable; for example `APS` and `webserver1`

3 **Perform the steps in "Configuring Web Server Routing" on page 135.**

Applying Post Deployment Application Settings

➤ To apply post deployment application settings:

1 **Enable Session Override for the deployed application,**

then change **Cookie path** to `/aps`.

2 **Restart the Web application server instance.**

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open:**

`http://HOSTNAME.EXAMPLE.COM:LISTEN_PORT/aps/APS`

2 **Review the output for your WebSphere Web application server instance in:**

`WAS_ROOT/profiles/PROFILE_NAME/logs`.

3 **Review the product logs in** `HYPERION_HOME/logs/aps`.

Deploying EPM Workspace

Configuring the Web Application Server Instance

➤ To configure the Web application server instance:

1 **Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".**

2 **Add the Java argument:**

`-Dhyperion.home=HYPERION_HOME`

See "WebSphere Environment Assumptions" on page 109.

3 **Add the following:**

a. Add the HYPERION_HOME environment variable to the configuration for your application server instance:

b. If necessary, add a library search path variable. See "Library Search Path Variables" on page 18.

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- If SAP is defined in Shared Services as a user directory, add

  `HYPERION_HOME/common/SAP/bin`

Windows example:

```
HYPERION_HOME   C:\Hyperion
PATH  C:\Hyperion\common\CSS\9.5.0.0\bin;C:\Hyperion\common\SAP\bin
```

UNIX and Linux example:

```
HYPERION_HOME      /opt/hyperion
LD_LIBRARY_PATH    /opt/hyperion/common/SAP/bin
```

Deploying workspace.war

➤ To deploy `workspace.war`:

1 **Deploy the following Web archive to the Web application server instance that you created:**

HYPERION_HOME`/products/Foundation/workspace/InstallableApps/workspace.war`

2 **During deployment, provide**

- **Context root**: `/workspace` (Case-sensitive; do not alter.)
- Enable **Precompile JavaServer Pages files**
- **Optional:** Provide a descriptive **Application Name**: for example, `Workspace`.
- **Map modules to servers**: Select the correct Web application server and Web server, if applicable; for example `Workspace` and `webserver1`

3 **Perform the steps in** "Configuring Web Server Routing" on page 135.

Optional: Deploying biplus_webservices.war

➤ To deploy `biplus_webservices.war`:

1 **Deploy the following Web archive to the instance that you created:**

HYPERION_HOME`/products/Foundation/workspace/InstallableApps/biplus_webservices.war`

2 **During deployment, provide: Application Name; for example,** `BIPlusWebServices`.

3 **Restart the instance.**

Applying Post Deployment Application Settings

➤ To apply post deployment application settings:

1 **Enable Session Override for the deployed application,**

then change **Cookie path** to `/workspace`

2 **Restart the Web application server instance.**

Customizing EPM Workspace Services Configuration Scripts

EPM Workspace Services include scripts that can be launched interactively to configure various part of the system. When the Manual option is selected during EPM Workspace deployment, the *DEPLOYMENT_HOME* variable declarations must be manually defined in

HYPERION_HOME/products/Foundation/workspace/bin/settrustedpass.bat|sh

➤ To declare the variable declarations:

1 **In a text editor, open:**

HYPERION_HOME/products/Foundation/workspace/bin/settrustedpass.bat|sh

2 **Replace occurrences of the *DEPLOYMENT_HOME* with:**

$J(trustedPass.deploymentHome)

where *DEPLOYMENT_HOME* is the file system path to the deployed EPM Workspace Web application.

Example:

set DEPLOYMENT_HOME=*PROFILE_HOME*/installedApps/*CELL*/workspace.ear/workspace.war

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open:**

http://*HOSTNAME.EXAMPLE.COM:LISTEN_PORT*/workspace/

2 **Review the output for your WebSphere Web application server instance in:**

WAS_ROOT/profiles/*PROFILE_NAME*/logs

3 **Review the product logs in** *HYPERION_HOME*/logs/workspace.

Deploying Performance Management Architect

Deploying Performance Management Architect Web

Configuring the Web Application Server Instance

➤ To configure the Web application server instance:

1 Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".

2 Add the Java argument:

`-Dhyperion.home=`*`HYPERION_HOME`*

See "WebSphere Environment Assumptions" on page 109.

3 If necessary, add a library search path variable. See "Library Search Path Variables" on page 18.

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- If SAP is defined in Shared Services as a user directory, add

  `HYPERION_HOME``/common/SAP/bin`

Windows example:

```
PATH C:\Hyperion\common\CSS\9.5.0.0\bin;C:\Hyperion\common\SAP\bin
```

UNIX and Linux example:

```
LD_LIBRARY_PATH    /opt/hyperion/common/SAP/bin
```

Deploying awb.war

➤ To deploy `awb.war`:

1 Deploy the following Web archive to the application server instance that you created:

`HYPERION_HOME``/products/Foundation/BPMA/AppServer/InstallableApps/awb.war`

2 During deployment, provide:

- **Context root:** /awb (Case-sensitive; do not alter.)
- Enable **Precompile JavaServer Pages files**
- **Application Name:** for example, `EPMAWeb`.
- **Map modules to servers**: Select the correct Web application server and Web server, if applicable; for example `EPMAWeb` and `webserver1`

3 Configure Web server routing. See "Configuring Web Server Routing" on page 135.

Applying Post Deployment Application Settings

➤ To apply post deployment application settings:

1 Enable **Session Override** for the deployed application,

then change **Cookie path** to `/awb`.

2 Restart the Web application server instance.

Deploying Performance Management Architect DataSync

Configuring the Web Application Server Instance

➤ To configure the Web application server instance:

1 Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".

2 Add the Java argument:

`-Dhyperion.home=`*`HYPERION_HOME`*

See "WebSphere Environment Assumptions" on page 109.

3 If necessary, add a library search path variable. See "Library Search Path Variables" on page 18.

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- If SAP is defined in Shared Services as a user directory, add

  `HYPERION_HOME``/common/SAP/bin`

Windows example:

```
PATH C:\Hyperion\common\CSS\9.5.0.0\bin;C:\Hyperion\common\SAP\bin
```

UNIX and Linux example:

```
LD_LIBRARY_PATH    /opt/hyperion/common/SAP/bin
```

Deploying DataSync.war

➤ To deploy `DataSync.war`:

1 Deploy the following Web archive to the application server instance that you created:

`HYPERION_HOME``/products/Foundation/BPMA/AppServer/InstallableApps/DataSync.war`

2 During deployment, provide

- **Context root:** /DataSync (Case-sensitive; do not alter.)
- **Application Name:** for example, `DataSync`.

3 Perform the steps in "Configuring Web Server Routing" on page 135.

Applying Post Deployment Application Settings

➤ To apply post deployment application settings:

1 Enable **Session Override** for the deployed application,

then change **Cookie path** to `/DataSync`.

2 Restart the Web application server instance.

Verifying Deployment

➤ To verify deployment:

1 Using a Web browser, open:

`http://HOSTNAME.EXAMPLE.COM:LISTEN_PORT/awb/`

2 Review the output for your WebSphere Web application server instance in *WAS_ROOT*`/profiles/`*PROFILE_NAME*`/logs`.

3 Review the product logs in *HYPERION_HOME*`/logs/epma`.

Deploying Calculation Manager

Configuring the Web Application Server Instance

➤ To configure the Web application server instance:

1 Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".

2 Add the Java argument:

`-Dhyperion.home=`*HYPERION_HOME*

See "WebSphere Environment Assumptions" on page 109.

3 Add an environment variable, HYPERION_HOME, to the configuration for you application server instance.

Windows example:

```
HYPERION_HOME      C:\Hyperion
```

UNIX and Linux example:

```
HYPERION_HOME      /opt/hyperion
```

Deploying calcmgr.ear

➤ To deploy `calcmgr.ear`:

1 **Deploy the following enterprise archive to the application server instance that you created:**

HYPERION_HOME`/products/Foundation/CALC/AppServer/InstallableApps/calcmgr.ear`

2 **During deployment, provide:**

- Enable **Precompile JavaServer Pages files**
- **Optional:** Provide a descriptive **Application Name**: for example, `Calcmg`
- **Map modules to servers**: Select the correct Web application server and Web server, if applicable; for example `Calcmg` and `webserver1`

3 **Perform the steps in** **"Configuring Web Server Routing" on page 135**.

Applying Post Deployment Application Settings

➤ To apply post deployment application settings:

1 **Enable Session Override for the deployed application,**

then change **Cookie path** to `/calcmgr`.

2 **Change the Class loader order for the deployed application to Classes loaded with application class loader first.**

Note: In older releases of WebSphere the administration console property to be edited was called **Class Loading and File Update Detection**, and the value to select was called **Parent Last**.

3 **Configure Web server routing. See** **"Configuring Web Server Routing" on page 135**

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open:**

`http://`*HOSTNAME.EXAMPLE.COM:LISTEN_PORT*`/calcmgr/`

2 **Review the output for your WebSphere Web application server instance in**

WAS_ROOT`/profiles/`*PROFILE_NAME*`/logs`.

3 **Review the product logs in** *HYPERION_HOME*`/logs/calcmgr`.

Deploying Financial Reporting

Configuring the Web Application Server Instance

➤ To configure the Web application server instance:

1 **Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".**

2 **Add the following:**

a. Add HYPERION_HOME environment variable to the configuration for your application server instance.

See "WebSphere Environment Assumptions" on page 109.

b. UNIX: Add an environment variable DISPLAY, and assign a valid X11 address as its value. See "Enabling an X Virtual Frame Buffer for Financial Reporting and Web Analysis" in the *Oracle Enterprise Performance Management System Installation and Configuration Guide*.

c. Add a library search path variable. See "Library Search Path Variables" on page 18.

Note: In Windows, PATH is set automatically.

Tip: UNIX: Execute: *HYPERION_HOME*`/products/biplus/InstallableApps/freporting_web.env`. Use script output as the library search path variable value.

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- If SAP is used as a data source or is defined in Shared Services as a user directory, add

  HYPERION_HOME`/common/SAP/bin`

- If Financial Management is used as a data source, add:

  HYPERION_HOME`/products/FinancialManagement/Client`
  HYPERION_HOME`/products/FinancialManagement/common`

- If Planning is used as a data source, add

  HYPERION_HOME`/common/ADM/Planning/`*VERSION*`/bin`

where *VERSION* = 9.5.0.0 if the version of Planning is this release (11.1.1.4). If Planning is a previous release, then *VERSION* is the version of the Planning you installed.

Windows example:

```
HYPERION_HOME    C:\Hyperion
```

UNIX and Linux example:

```
HYPERION_HOME    /opt/hyperion
ESSLANG          English_UnitedStates.Latin1@Binary
ARBORPATH        /opt/hyperion/common/EssbaseRTC/9.5.0.0
ESSBASEPATH      /opt/hyperion/common/EssbaseRTC/9.5.0.0
ESS_ES_HOME      /opt/hyperion/products/biplus/bin/EssbaseJAPI

DISPLAY      xvfb-host.example.com:99.0

LD_LIBRARY_PATH    /opt/hyperion/products/biplus/bin:
/opt/hyperion/common/EssbaseRTC/9.5.0.0/bin:
/opt/hyperion/common/ADM/9.5.0.0/Essbase/9.5.0.0/bin:
/opt/hyperion/common/SAP/bin:
/opt/hyperion/products/FinancialManagement/Client:
/opt/hyperion/products/FinancialManagement/common
```

Deploying HReports.ear

➤ To deploy `HReports.ear`:

1 **Deploy the following enterprise archive to the application server instance that you created:**

HYPERION_HOME`/products/biplus/InstallableApps/HReports.ear`

2 **During deployment, provide:**

- Enable **Precompile JavaServer Pages files**
- **Optional:** Provide a descriptive **Application Name**: for example, `FinancialReporting`
- **Map modules to servers**: Select the correct Web application server and Web server, if applicable; for example `FinancialReporting` and `webserver1`

3 **Perform the steps in "Configuring Web Server Routing" on page 135.**

Applying Post Deployment Application Settings

➤ To apply post deployment application settings:

1 **Enable Session Override for the deployed application,**

then change **Cookie path** to `/hr`.

2 **Restart the Web application server instance.**

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open:**

`http://`*`HOSTNAME.EXAMPLE.COM:LISTEN_PORT`*`/hr/status.jsp`

2 **Review the output for your WebSphere Web application server instance in** *`WAS_ROOT`*`/profiles/`*`PROFILE_NAME`*`/logs`.

3 **Review the product logs in** *`HYPERION_HOME`*`/logs/BIPlus`.

Deploying Web Analysis

Configuring the Web Application Server Instance

➤ To configure the Web application server instance:

1 **Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".**

2 **Add these Java arguments:**

- `-DHYPERION_HOME=`*`HYPERION_HOME`*
- `-DBIPLUS_HOME=`*`HYPERION_HOME`*`/products/biplus`

  See "WebSphere Environment Assumptions" on page 109.

3 **Add the following CLASSPATH entries:**

`HYPERION_HOME``/common/CLS/9.5.0.0/lib/cls-9_5_0.jar`
`HYPERION_HOME``/common/JDBC/DataDirect/3.7/lib/hyjdbc.jar`
`HYPERION_HOME``/common/SAP/lib`

Note: Separate entries with semi-colons (;) for Windows, colons (:) for UNIX.

4 **Add the following:**

a. Add the following environment variables to the configuration for your application server instance:

- HYPERION_HOME
- BIPLUS_HOME
- ESSLANG
- ARBORPATH

- ESSBASEPATH
- ESS_ES_HOME
- ICU_DATA
- DISPLAY (UNIX): Add an environment variable DISPLAY, and assign a valid X11 address as its value. See "Enabling an X Virtual Frame Buffer for Financial Reporting and Web Analysis" in the *Oracle Enterprise Performance Management System Installation and Configuration Guide*.

b. Add a library search path variable. See "Library Search Path Variables" on page 18.

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- If SAP is used as a data source or is defined in Shared Services as a user directory, add:

  HYPERION_HOME/common/SAP/bin

- If Financial Management is used as a data source, add:

  HYPERION_HOME/products/FinancialManagement/Client
  HYPERION_HOME/products/FinancialManagement/common

Windows example:

```
HYPERION_HOME  C:\Hyperion
BIPLUS_HOME  C:\Hyperion\products\biplus
ESSLANG  English_UnitedStates.Latin1@Binary
ARBORPATH   C:\Hyperion\common\EssbaseRTC\9.5.0.0
ESSBASEPATH  C:\Hyperion\common\EssbaseRTC\9.5.0.0
ESS_ES_HOME  C:\Hyperion\products\biplus\bin\EssbaseJAPI
ICU_DATA  C:\Hyperion\common\ADM\9.5.0.0\Essbase\9.5.0.0\bin\HssEssDriver

PATH  C:\Hyperion\products\biplus\bin;
C:\Hyperion\common\EssbaseRTC\9.5.0.0\bin;
C:\Hyperion\common\ADM\9.5.0.0\Essbase\9.5.0.0\bin;
C:\Hyperion\common\SAP\bin;
C:\Hyperion\products\FinancialManagement\Client;
C:\Hyperion\products\FinancialManagement\Common
```

UNIX and Linux example:

```
HYPERION_HOME  /opt/hyperion
BIPLUS_HOME  /opt/hyperion/products/biplus
ESSLANG  English_UnitedStates.Latin1@Binary
ARBORPATH  /opt/hyperion/common/EssbaseRTC/9.5.0.0
ESSBASEPATH  /opt/hyperion/common/EssbaseRTC/9.5.0.0
ESS_ES_HOME  /opt/hyperion/products/biplus/bin/EssbaseJAPI
ICU_DATA  /opt/hyperion/common/ADM/9.5.0.0/Essbase/9.5.0.0/bin/HssEssDriver

DISPLAY     xvfb-host.example.com:99.0

LD_LIBRARY_PATH  /opt/hyperion/products/biplus/bin:
/opt/hyperion/common/EssbaseRTC/9.5.0.0/bin:
/opt/hyperion/common/ADM/9.5.0.0/Essbase/9.5.0.0/bin:
```

```
/opt/hyperion/common/SAP/bin:
/opt/hyperion/products/FinancialManagement/Client:
/opt/hyperion/products/FinancialManagement/common
```

Deploying WebAnalysis.ear

➤ To deploy `WebAnalysis.ear`:

1 **Deploy the following enterprise archive to the application server instance that you created:**

HYPERION_HOME`/products/biplus/InstallableApps/WebAnalysis.ear`

2 **During deployment, provide:**

- Enable **Precompile JavaServer Pages files**
- **Optional:** Provide a descriptive **Application Name**: for example, `WebAnalysis`
- **Map modules to servers**: Select the correct Web application server and Web server, if applicable; for example `WebAnalysis` and `webserver1`

3 **Perform the steps in** **"Configuring Web Server Routing" on page 135**.

Applying Post Deployment Application Settings

➤ To apply post deployment application settings:

1 **Enable Session Override for the deployed application,**

then change **Cookie path** to `/WebAnalysis`.

2 **Restart the Web application server instance.**

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open:**

`http://`*HOSTNAME.EXAMPLE.COM:LISTEN_PORT*`/WebAnalysis/`

2 **Review the output for your WebSphere Web application server instance in** *WAS_ROOT*`/profiles/`*PROFILE_NAME*`/logs`.

3 **Review the product logs in** *HYPERION_HOME*`/logs/BIPlus`.

Deploying Planning

- "Verifying Deployment" on page 129

Configuring the Web Application Server Instance

➤ To configure the Web application server instance:

1 **Start the `RMIRegistry` for Oracle Hyperion Planning, Fusion Edition using:**

- **Windows:** Windows service
- **UNIX and Linux:** Execute: *`HYPERION_HOME`*`/common/RMI/HyperionRMIRegistry`

2 **Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".**

3 **Add the following CLASSPATH entry:**

`HYPERION_HOME``/common/XML/JDOM/0.9.0/hit-common.jar`
`HYPERION_HOME``/common/XML/JDOM/0.9.0/hit-config.jar`

See "WebSphere Environment Assumptions" on page 109.

4 **Add the following:**

a. Add the following environment variables to the configuration for your application server instance:

- HYPERION_HOME
- PLANNING_HOME

b. Add a library search path variable. See "Library Search Path Variables" on page 18.

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- If SAP is defined in Shared Services as a user directory, add

  `HYPERION_HOME``/common/SAP/bin`

Windows example:

```
HYPERION_HOME    C:\Hyperion
PLANNNING_HOME   C:\Hyperion\products\Planning
PATH C:\Hyperion\common\EssbaseRTC\9.5.0.0\bin;C:\hyperion\common\SAP\bin
```

UNIX and Linux example:

```
HYPERION_HOME    /opt/hyperion
PLANNNING_HOME   /opt/hyperion/products/Planning/lib
LD_LIBRARY_PATH="/opt/hyperion/common/EssbaseRTC/9.5.0.0/bin:/opt/hyperion/
products/Planning/lib::/opt/hyperion/common/SAP/bin:$LD_LIBRARY_PATH";export
LD_LIBRARY_PATH
```

Deploying HyperionPlanning.ear

➤ To deploy `HyperionPlanning.ear`:

1 **Deploy the following enterprise archive to the application server instance that you created:**

HYPERION_HOME`/products/Planning/AppServer/InstallableApps/common/HyperionPlanning.ear`

2 **During deployment, provide:**

- Enable **Precompile JavaServer Pages files**
- **Optional:** Provide a descriptive **Application Name**: for example, `HyperionPlanning`
- **Map modules to servers**: Select the correct Web application server and Web server, if applicable; for example `HyperionPlanning` and `webserver1`

3 **Perform the steps in** **"Configuring Web Server Routing" on page 135**.

Applying Post Deployment Application Settings

➤ To apply post deployment application settings:

1 **Enable Session Override for the deployed application,**

then change **Cookie path** to `/HyperionPlanning`.

2 **Change the Class loader order for the deployed application to Classes loaded with application class loader first.**

Note: In older releases of WebSphere the administration console property to be edited was called **Class Loading and File Update Detection**, and the value to select was called Parent Last.

3 **Restart the Web application server instance.**

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open:**

`http://`*HOSTNAME.EXAMPLE.COM:LISTEN_PORT*`/HyperionPlanning/`

2 **Review the output for your WebSphere Web application server instance in**

WAS_ROOT`/profiles/`*PROFILE_NAME*`/logs`.

3 **Review the product logs in** *HYPERION_HOME*`/logs/Planning`.

Deploying Performance Scorecard

Configuring the Web Application Server Instance

➤ To configure the Web application server instance:

1 **Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".**

2 **Add these Java arguments:**

- `-Dhyperion.home=`*`HYPERION_HOME`*
- `-DESS_ES_HOME=`*`HYPERION_HOME`*`/common/EssbaseRTC/9.5.0.0.`

  See "WebSphere Environment Assumptions" on page 109.

3 **Add the following:**

a. Add these environment variables to the configuration for your application server instance:

- HYPERION_HOME
- ARBORPATH
- ESSBASEPATH
- ESSLANG

b. Add a library search path variable. See "Library Search Path Variables" on page 18.

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- If SAP is defined in Shared Services as a user directory, add

  `HYPERION_HOME``/common/SAP/bin`

Windows example:

```
HYPERION_HOME  c:\Hyperion
ARBORPATH      c:\Hyperion\common\EssbaseRTC\9.5.0.0
ESSBASEPATH    c:\Hyperion\common\EssbaseRTC\9.5.0.0
ESSLANG        English_UnitedStates.Latin1@Binary

PATH  C:\Hyperion\common\EssbaseRTC\9.5.0.0\bin;
C:\Hyperion\common\ADM\9.5.0.0\Essbase\9.5.0.0\bin;
C:\Hyperion\common\SAP\bin
```

UNIX and Linux example:

```
HYPERION_HOME   /opt/hyperion
ARBORPATH       /opt/hyperion/common/EssbaseRTC/9.5.0.0
ESSBASEPATH     /opt/hyperion/common/EssbaseRTC/9.5.0.0
ESSLANG         English_UnitedStates.Latin1@Binary

LD_LIBRARY_PATH  /opt/common/EssbaseRTC/9.5.0.0/bin:
opt/common/ADM/9.5.0.0/Essbase/9.5.0.0/bin:
opt/common/SAP/bin
```

Deploying HPSWebReports.war

➤ To deploy `HPSWebReports.war`:

1 **Deploy the following Web archive to the application server instance that you created:**

HYPERION_HOME`/products/PerformanceScorecard/AppServer/InstallableApps/common/webapps/configured/HPSWebReports.war`

2 **During deployment, provide**

- **Context root:** `/HPSWebReports` (Case-sensitive; do not alter.)
- Enable **Precompile JavaServer Pages files**
- **Optional:** Provide a descriptive **Application Name:** for example, `Scorecard`
- **Map modules to servers**: Select the correct Web application server and Web server, if applicable; for example `Scorecard` and `webserver1`

3 **Perform the steps in** **"Configuring Web Server Routing" on page 135**.

Deploying HPSAlerter.war

➤ To deploy `HPSAlerter.war`:

1 **Deploy the following Web archive to the application server instance that you created:**

HYPERION_HOME`/products/PerformanceScorecard/AppServer/InstallableApps/common/webapps/configured/HPSAlerter.war`

2 **During deployment, provide**

- **Context root:** `/HPSAlerter` (Case-sensitive; do not alter)
- Enable **Precompile JavaServer Pages files**
- **Optional:** Provide a descriptive **Application Name:** for example, `Alerter`
- **Map modules to servers**: Select the correct Web application server and Web server, if applicable; for example `Alerter` and `webserver1`

3 **Perform the steps in** **"Configuring Web Server Routing" on page 135**.

Applying Post Deployment Application Settings

➤ To apply post deployment application settings:

1 **Enable Session Override for the deployed application,**

then change **Cookie path** to `/HPSWebReports` or `/HPSAlerter`.

2 **Restart the Web application server instance.**

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open:**

`http://`*`HOSTNAME.EXAMPLE.COM:LISTEN_PORT`*`/workspace/`

2 **Access the Performance Scorecard module using the EPM Workspace navigation menu.**

3 **Review the output for your WebSphere Web application server instance in**

`WAS_ROOT``/profiles/`*`PROFILE_NAME`*`/logs`.

4 **Review the product logs in** *`HYPERION_HOME`*`/logs/hps`.

Deploying Profitability and Cost Management

Creating Data Sources in WebSphere

If you are using IBM WebSphere 6.1 with Profitability and Cost Management, you need to manually create the data source for your Profitability and Cost Managementapplication before deploying to WebSphere.

➤ To create a data source for Profitability application using WebSphere:

1 **From the WebSphere Administrative Console, select Resources, then JDBC, and then Data sources.**

2 **Select an appropriate scope for your cell or node.**

3 **On the Create a data source screen, click New.**

Step 1 of the Create a Data Source wizard is displayed.

4 **Under Data source name, enter the display name for the new data source, for example, hpm.**

5 Under **JNDI name**, enter **jdbc/hpm**.

6 **Optional:** Under **Component-managed authentication alias**, select a component managed authentication alias.

If no authentication alias has been defined, you can define and assign the alias as follows:

a. From the WebSphere main menu, select **Security**, then **Secure administration, applications, and infrastructure**.

b. Under **Authentication**, select **JAAS — J2C authentication data** (Java Authentication and Authorization Service).

c. Enter the following information to specify the user credentials used in your database for the Java 2 connector security to use:

- Alias (This alias is displayed under the **Component-managed authentication alias...** drop-down list when creating the new data source for the JDBC provider.
- User ID
- Password
- Description (optional)

d. Click **OK** to create the new authentication alias.

e. Open the data source that was just created, and select the new authentication alias from the **Component-managed authentication alias** drop-down list.

7 Click **Next**.

8 On the **Create new JDBC Provider** page, select the **Database type** for the new JDBC provider.

9 Select the **Provider type**.

10 Under **Implementation type**, select **Connection data pool source**.

The name field is populated with the Provider type.

11 **Optional:** Enter a brief **Description**.

12 Click **Next**.

13 On the **Enter database specific properties for the data source** screen, enter the **Database name**, **Server name**, and **Port number**.

14 Click **Next**.

The Summary window is displayed, showing all the selections for the data source.

15 Verify the selections in the Summary window.

16 Click **Finish**.

Configuring the Web Application Server Instance

➤ To configure the Web application server instance:

1 Complete the steps in Chapter 2, "Setting Up Your System for Manual Deployment: Required Steps".

See "WebSphere Environment Assumptions" on page 109.

2 **Add the Java argument:**

`-Dhyperion.home=`*`HYPERION_HOME`*

3 **Add the following:**

a. Environment variable: HYPERION_HOME

b. If necessary, add a library search path variable. See "Library Search Path Variables" on page 18.

Notes:

When creating a library search path variable:

- Separate entries with semi-colons (;) for Windows, colons (:) for UNIX
- If SAP is defined in Shared Services as a user directory, add

  `HYPERION_HOME``/common/SAP/bin`

Windows example:

```
HYPERION_HOME   C:\Hyperion
PATH C:\Hyperion\common\SAP\bin
```

UNIX and Linux example:

```
HYPERION_HOME   /opt/hyperion
LD_LIBRARY_PATH   /opt/hyperion/common/SAP/bin
```

Deploying profitability.war

➤ To deploy `profitability.war`:

1 **Deploy the following enterprise archive to the application server instance that you created:**

`HYPERION_HOME``/products/Profitability/AppServer/InstallableApps/common/profitability.war`

2 **During deployment, provide:**

- **Context root:** `/profitability` (Case-sensitive; do not alter.)
- Enable **Precompile JavaServer Pages files**
- **Optional:** Provide a descriptive **Application Name:** for example, `Profitability`
- **Map modules to servers**: Select the correct Web application server and Web server, if applicable; for example `Profitability` and `webserver1`

3 **Configure Web server routing. See "Configuring Web Server Routing" on page 135.**

Applying Post Deployment Application Settings

➤ To apply post deployment application settings:

1 **Enable Session Override for the deployed application,**

then change **Cookie path** to `/profitability`.

2 **Restart the Web application server instance.**

Verifying Deployment

➤ To verify deployment:

1 **Using a Web browser, open:**

`http://`*`HOSTNAME.EXAMPLE.COM:LISTEN_PORT`*`/Profitability/ping.jsp`

2 **Review the output for your WebSphere Web application server instance in**

`WAS_ROOT``/profiles/`*`PROFILE_NAME`*`/logs`.

3 **Review the product logs in** *`HYPERION_HOME`*`/logs/profitability`.

Configuring Web Server Routing

This section describes how to customize the WebSphere Web server plug-in configuration for IBM HTTP Server (IHS), Apache and IIS.

Prerequisites

To enable the WebSphere Web server plug-in for your Web server, follow the instructions in the *IBM WebSphere Information Center*: "Installing Web Server Plug-ins" section.

Tip: Apache HTTP Server is included with the EPM System, and is installed in *`HYPERION_HOME`*`/common/httpServers/Apache`.

Modifying plugin-cfg.xml

You must reapply the customizations described in this section each time `plugin-cfg.xml` is regenerated.

Shared Services

If using a Web server to proxy requests to Shared Services, you must change the `AcceptAllContent` attribute from false to true in WebSphere's `plugin-cfg.xml`; for example,

```
AcceptAllContent="true"
```

EPM Workspace

Using the dynamic content forward paths identified in Appendix A, "Forward Paths For Web Server Routing" as reference, verify that the directive that instructs the Web server to proxy all requests to the context `/workspace` is replaced with new directives that instruct the Web server to proxy only requests for dynamic content.

Example:

```
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/workspace/*"/
>
```

with:

```
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/workspace/
cdsrpc"/>
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/workspace/
administration/*"/>
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/workspace/
browse/*"/>
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/workspace/
personalpages/*"/>
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/workspace/
viewmanager/*"/>
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/workspace/
jobmanager/*"/>
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/workspace/
ihtml/*"/>
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/workspace/
dataaccess/*"/>
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/workspace/
logon"/>
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/workspace/
logon/*"/>
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/workspace/
prefs"/>
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/workspace/js/
*"/>
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/workspace/
modules/*"/>
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/workspace/
resources/*"/>
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/workspace/
conf/*"/>
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/workspace/
media/*"/>
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/workspace/
BPMContext"/>
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/workspace/
wsrp4j/*"/>
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/workspace/
ResourceProxy"/>
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/workspace/
PDFView"/>
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/workspace/"/>
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/workspace/
*.jsp"/>
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/workspace/
```

```
*.jsv"/>
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/workspace/
*.jsw"/>
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/workspace/
DynamicHelp" />
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/workspace/
search" />
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/workspace/
servlet/*" />
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/workspace/
portletservlet/*" />
```

Web Analysis

Using the dynamic content forward paths identified in Appendix A, "Forward Paths For Web Server Routing" as reference, replace the directive that instructs the Web server to proxy all requests to the context `/WebAnalysis` with new directives that instruct the Web server to proxy only requests for dynamic content.

Example:

```
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/WebAnalysis/
*"/>
```

with:

```
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/WebAnalysis/
ServerConsole/*" />
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/WebAnalysis/
templates/*" />
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/WebAnalysis/
hfmtemplates/*" />
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/WebAnalysis/
hitemplates/*" />
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/WebAnalysis/
processor" />
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/WebAnalysis/
servlet/*" />
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/WebAnalysis/
modules/*" />
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/WebAnalysis/
resources/*" />
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/WebAnalysis/
DirectoryServlet" />
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/WebAnalysis/
config" />
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/WebAnalysis/
wa_javadocs" />
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/
WebAnalysis/" />
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/WebAnalysis/
*.jsp" />
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/WebAnalysis/
*.xml" />
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/WebAnalysis/
*.jar" />
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/WebAnalysis/
```

```
*.exe" />
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/WebAnalysis/
Config" />
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/WebAnalysis/
portletservlet/*" />
<Uri AffinityCookie="JSESSIONID" AffinityURLIdentifier="jsessionid" Name="/WebAnalysis/
portlettemplates/*" />
```

Configuring Access To Static Content

To improve performance, and to reduce load on the Web application server, static content and online help are served by the Web server for EPM Workspace and Oracle's Hyperion® Web Analysis.

Note: If your Web server is not on the same machine where EPM Workspace was installed, you must manually copy the `workspace_static` and `WebAnalysis_static` folders from *HYPERION_HOME*`/products/Foundation/workspace/AppServer/InstalledApps` to your Web server machine.

EPM Workspace

IBM HTTP Server and Apache

Using a text editor, open your Web server's configuration file (`httpd.conf`), then add the following directives, replacing *HYPERION_HOME* as needed:

```
# Change modules/mod_expires.so as needed; for example, libexec/mod_expires on UNIX
<IfModule !expires_module>
  LoadModule expires_module modules/mod_expires.so
</IfModule>
# Images are unlikely to change, so force the browser to cache them. 3 months is the
default.
<LocationMatch  /workspace/(themes|thirdparty|img|images|css|media|docs)/.*/.(gif|jpeg|
jpg|png)$ >
 ExpiresDefault "now"
 ExpiresActive on
 ExpiresByType image/gif "now plus 3 month"
 ExpiresByType image/jpg "now plus 3 months"
 ExpiresByType image/jpeg "now plus 3 months"
 ExpiresByType image/png "now plus 3 months"
</LocationMatch>

AliasMatch /workspace/(resources|css|docs|images|img|SmartView|themes|thirdparty|
wsmedia|zeroadmin|CrystalBall)/(.*) HYPERION_HOME/products/Foundation/workspace/
AppServer/InstalledApps/workspace_static/$1/$2
Alias /wsmedia "HYPERION_HOME/products/Foundation/workspace/AppServer/InstalledApps/
workspace_static/wsmedia"
Alias /InsightInstaller "HYPERION_HOME/products/Foundation/workspace/AppServer/
InstalledApps/workspace_static/zeroadmin/component/Insight"
```

IIS

➤ To configure IIS:

1 **Create a new Virtual Directory with the attributes:**

- **Alias:** `workspace` (case-sensitive)
- **Path:** *HYPERION_HOME*`/products/Foundation/workspace/AppServer/InstalledApps/workspace_static`
- **Permissions:** Read

2 **Create a new Virtual Directory with the attributes:**

- **Alias:** `wsmedia` (case-sensitive)
- **Path:** *HYPERION_HOME*`/products/Foundation/workspace/AppServer/InstalledApps/workspace_static/wsmedia`
- **Permissions:** Read

3 **Create a new Virtual Directory with the attributes:**

- **Alias:** `InsightInstaller` (case-sensitive)
- **Path:** *HYPERION_HOME*`/products/Foundation/workspace/AppServer/InstalledApps/workspace_static/zeroadmin/component/Insight`
- **Permissions:** Read

4 **Configure MIME types for `InsightInstaller` virtual directory:**

a. Right click the `InsightInstaller` virtual directory and select Properties.

b. Select HTTP Headers tab and click MIME Types.

c. Add following Extension / MIME types pairs:

```
.exe / application/octet-stream
.ex_ / application/octet-stream
.msi / application/octet-stream
.boot / application/octet-stream
.hdr / application/octet-stream
.ini / application/octet-stream
.xpi / application/octet-stream
```

Web Analysis

IBM HTTP Server and Apache

Using a text editor, open your Web server's configuration file (`httpd.conf`), then add the following directives, replacing *HYPERION_HOME* as needed:

```
AliasMatch /WebAnalysis/(images|js|resources|themes)/(.*) HYPERION_HOME/products/
Foundation/workspace/AppServer/InstalledApps/WebAnalysis_static/$1/$2
```

IIS

To configure IIS, Create a new Virtual Directory with the attributes:

- **Alias:** `WebAnalysis` (case-sensitive)
- **Path:** *`HYPERION_HOME`*`/products/Foundation/workspace/AppServer/InstalledApps/WebAnalysis_static`
- **Permissions:** Read, Run scripts, Execute

Configuring Financial Management with a Web Server

Configuring Financial Management with Apache Web Server or IBM HTTP Server

To configure Financial Management with Apache Web Server or IBM HTTP Server: using a text editor, open the Web server's configuration file (`httpd.conf`), and add the following directives, replacing *`HFM_HOST`* and *`HFM_PORT`* with real values:

```
 #Change modules/mod_proxy.so as needed; e.g. libexec/mod_proxy on UNIX
<IfModule !mod_proxy.c>
LoadModule proxy_module modules/mod_proxy.so
</IfModule>
<IfModule !mod_proxy_connect.c>
LoadModule proxy_connect_module modules/mod_proxy_connect.so
</IfModule>
<IfModule !mod_proxy_http.c>
LoadModule proxy_http_module modules/mod_proxy_http.so
</IfModule>
ProxyRequests Off
#For Oracle HTTP Server, the following line:
ProxyPreserveHost On
ProxyPass /hfm http://HFM_HOST:HFM_PORT/hfm
ProxyPassReverse /hfm http://HFM_HOST:HFM_PORT/hfm
```

where <HFM_HOST> is the Financial Management host machine and *`HFM_PORT`* is the default port, 80.

Using Only IIS with Financial Management

For using only IIS (for example, if Apache is not used at all): You must configure the IIS used by Oracle Hyperion Financial Management, Fusion Edition as the front-end Web server for EPM Workspace. You can have the EPM Workspace Web application server running on a separate machine, but you must install a copy of the EPM Workspace component onto the machine running IIS. The EPM Workspace installation includes files that are served by the Web server, such as the help files.

After installing Oracle Enterprise Performance Management Workspace, Fusion Edition on the IIS machine, use Life Cycle Management to change the Web server hostname.

6 Clustering EPM System Web Applications

In This Chapter

This chapter assumes that you are familiar with Java application server administration and clustering. If you are unfamiliar with these tasks, Oracle urges you to seek appropriate technical assistance before attempting to cluster a EPM System Web application.

For instructions on clustering EPM System services and products with proprietary application servers, see the *Oracle Hyperion Enterprise Performance Management System High Availability Guide*.

EPM System Web Application Clustering Prerequisites

Complete these tasks before setting up a cluster for a EPM System Web application:

- Set up a Java application server on each node that the cluster will include. See the Java application server documentation for instructions.
- Install the EPM System product on each node that the cluster will include. Install to the same file system location on each machine. Using the same file system path on each physical machine in a cluster is important so that these environment variables can be set once for the entire cluster, rather than set and customized for each node in the cluster:
  - All OS—`CLASSPATH` and `PATH`
  - UNIX—`LD_LIBRARY_PATH`, `LIBPATH`, or `SHLIB_PATH`

  See the *Oracle Enterprise Performance Management System Installation and Configuration Guide*.
- Configure the EPM System product for manual deployment by selecting the Manual Deployment option when you run Oracle's Hyperion Enterprise Performance Management System Configurator for each installation.

Setting Up an EPM System Web Application Cluster

➤ To set up a EPM System Web application cluster:

1 **Create the cluster.**

See the documentation for your Java application server.

2 **Manually deploy the archive (EAR or WAR) file for the EPM System product to the cluster.**

By default, the archive file is in the *HYPERION_HOME*/deployments directory.

3 **Complete any additional procedures for the specific EPM System product.**

See the remaining topics in this chapter for information specific to individual EPM System Web applications. Consult the Java application server vendor documentation for additional information about setting up Java application server clusters.

Clustering a EPM System Web application may require additional steps, depending on your Java application server and your environment. See your Java application server documentation. For Oracle Hyperion Enterprise Performance Management System Web application clustering issues that cannot be resolved using the Java application server documentation, Oracle consulting services are available.

Product-Specific Clustering Information

The following sections provide information about deploying specific products to Java application server clusters.

Profitability and Cost Management Clusters

Profitability and Cost Management cache clustering support is driven by the cache clustering support of TopLink Essentials, which is bundled with Oracle Hyperion Profitability and Cost Management, Fusion Edition. TopLink also offers cache coordination where changes made in one node can be synchronized, replicated, or invalidated across multiple nodes of the same application forming a cluster or grid. For more information, see these documents about TopLink:

- http://www.oracle.com/technology/tech/java/newsletter/articles/toplink/toplink_caching_locking.html
- http://www.oracle.com/technology/products/ias/toplink/JPA/essentials/toplink-jpa-extensions.html#TopLinkCaching

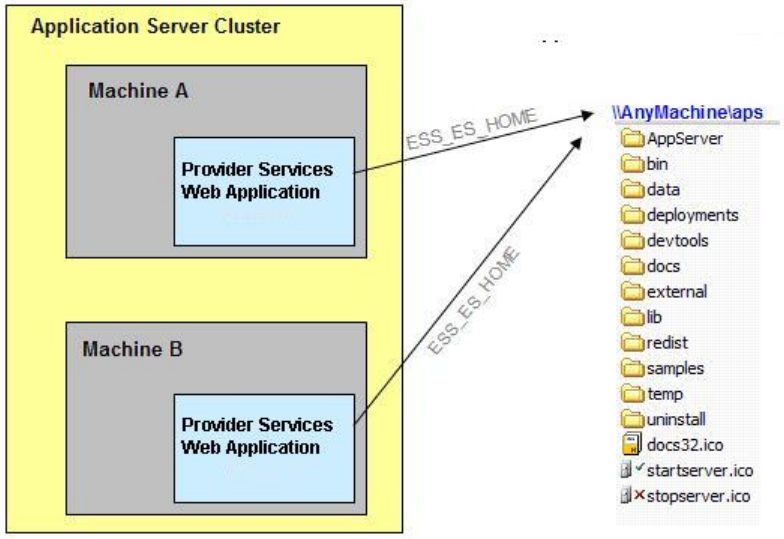
Provider Services Clusters

Clustering Provider Services for load balancing involves deploying multiple instances of Provider Services to handle requests from Oracle Hyperion Smart View for Office, Fusion Edition, Java

API, or XMLA clients. When you connect to Provider Services, client requests are handled by the same Provider Services instance until you disconnect.

For vertical or horizontal clustering, you need only one Provider Services installation. All application server instances under the cluster where the Provider Services Web application (`aps.war`) is deployed must refer to the same Provider Services file resources, so the *`PROVIDER_SERVICES_HOME`* file system must be shared and accessible with read-write permissions across all machines in a horizontal cluster, as illustrated in Figure 1. All provider instances can then refer to common configuration and data resources under the Provider Services installation.

Figure 1 Provider Services Home Directory Shared file System



When you create a Provider Services cluster through Oracle Essbase Administration Services Console, all Oracle Essbase servers in the cluster must be configured for the same type of authentication. The servers can be configured for Oracle's Hyperion® Shared Services or native authentication.

All application servers or instances under the cluster must be started by the user who installed Provider Services, because read-write permissions for file resources exist only for that user by default. So, in the horizontal clustering example in Figure 1, if user XYZ installs Provider Services on Machine A, you must ensure that user XYZ starts the application servers on Machine A and Machine B.

Before starting the application server cluster where you have deployed `aps.war`, you must make these changes to the Java Virtual Machine (JVM) options:

- For each application server instances under the cluster, modify the JVM options to add –`DESS_ES_HOME=`*`PROVIDER_SERVICES_HOME`*.

  For example, `-DESS_ES_HOME=C:/Hyperion/products/Essbase/aps`for vertical clustering, or `-DESS_ES_HOME=//machine_x/aps`.

- To enable separate logging for each Provider Services instance of the cluster, define a unique log file name for each instance by adding this JVM option to define a unique log file name:

`-DPROVIDER_ID=`*`Provider_Log_File_Name`*

For example, if there are two instances under the cluster, Provider1 and Provider2, edit the JVM options for Provider1 to add `-DPROVIDER_ID=provider_1.log`, and edit the options for Provider2 to add `-DPROVIDER_ID=provider_2.log`.

The specified log is created in *`PROVIDER_SERVICES_HOME`*`/bin`.

For instructions on setting up Provider Servicesclusters on a specific Java application server, see the Java application server documentation.

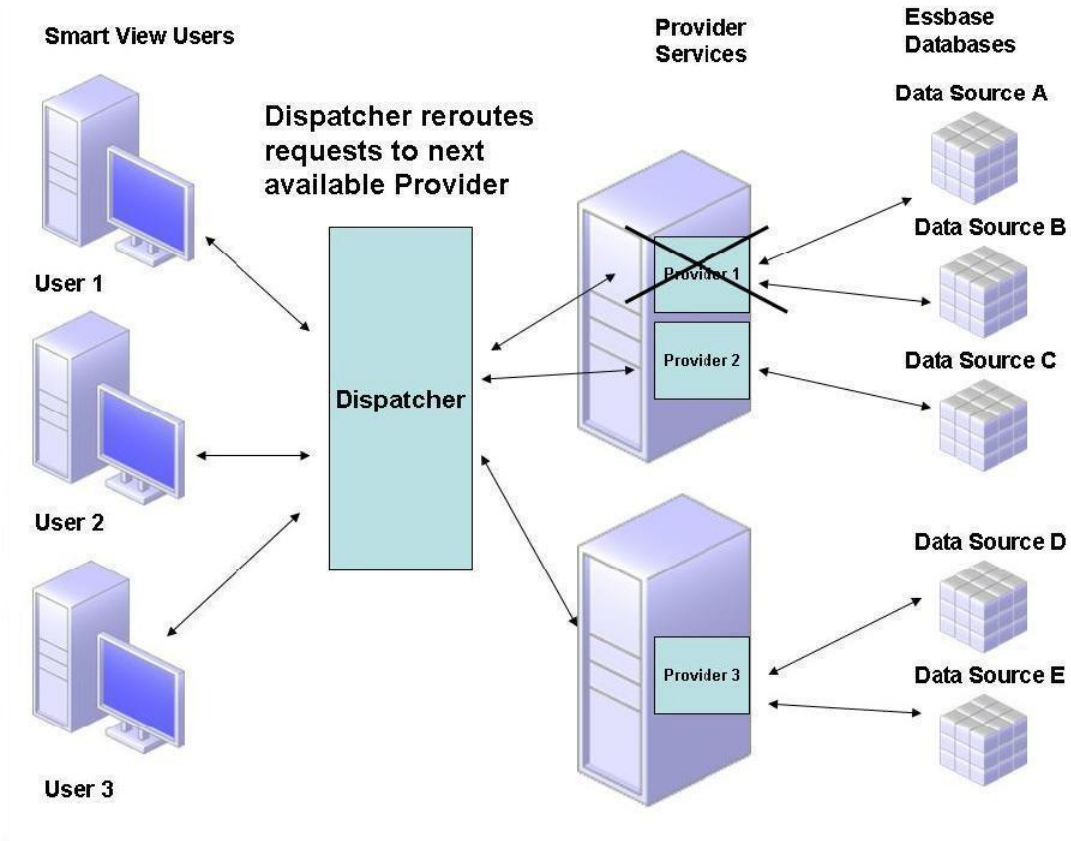
The following sections provide Provider Services clustering examples using Smart View.

Clustering Provider Services for Failover

Provider Services clusters implement "sticky" sessions: After a client user session is established with a Provider Services instance, further requests from the client are handled by the same Provider Services instance until the user disconnects. When a Provider Services instance within the cluster fails, and you receive an invalid session error, you must disconnect from the session, log on again using the same credentials, and resubmit the query. The dispatcher then connects to an available Provider Services instance in the cluster, using the same Java application server URL. A new session with a healthy Provider Services instance within the cluster is established. Figure 2 shows an example of Provider Services clustering for failover. In the example, a dispatcher reroutes requests from three Smart View users to the next available Provider Services provider when the original provider is offline.

provider

Figure 2 Provider Services Cluster Failover

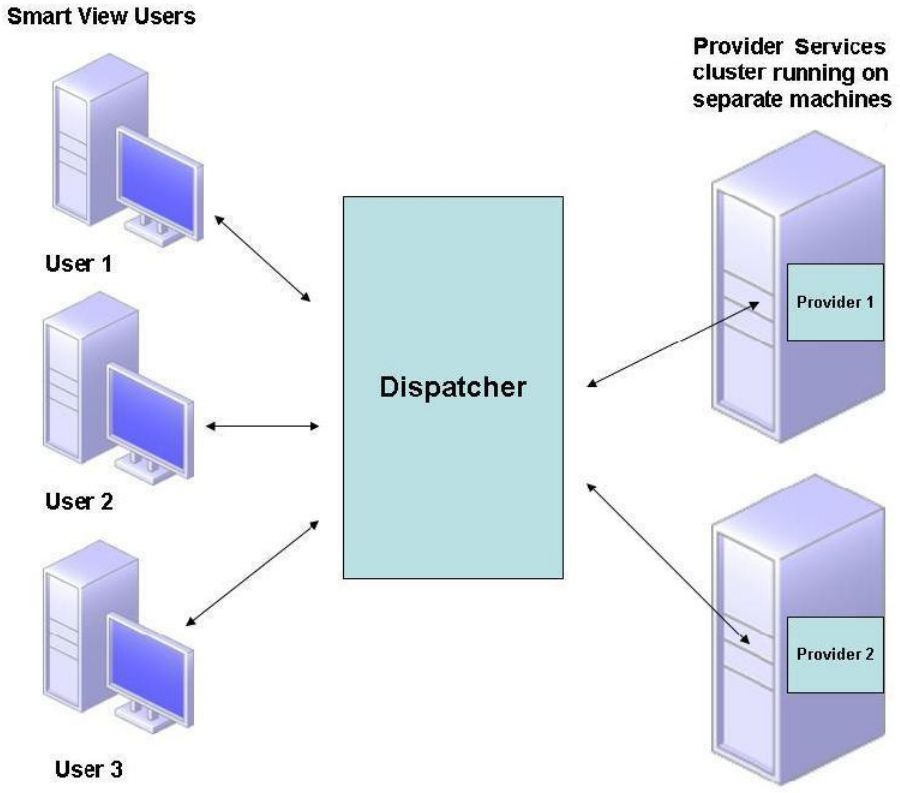


Horizontal Provider Services Clustering

Horizontal clustering deploys Java application server instances across multiple machines.

In Figure 3, which illustrates horizontal Provider Services clustering, a dispatcher distributes Smart View client requests between two providers running on different machines.

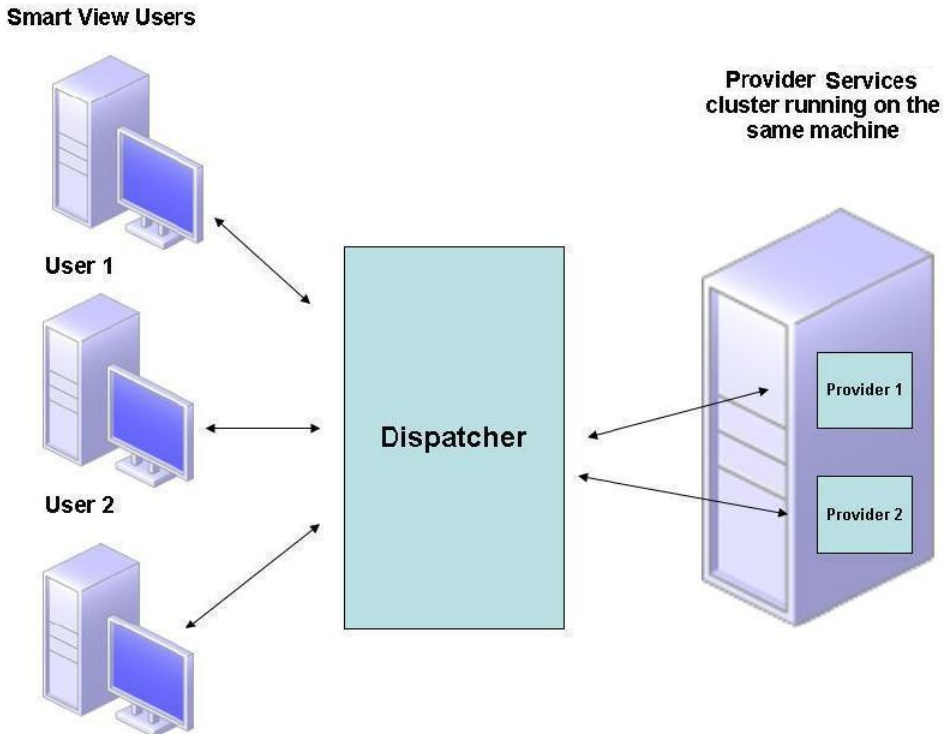
Figure 3 Horizontal Provider Services Cluster



Vertical Provider Services Clustering

Vertical clustering deploys multiple Java application server instances on one machine. In Figure 4, which illustrates vertical Provider Services clustering, a dispatcher distributes Smart View client requests between two providers running on the same machine.

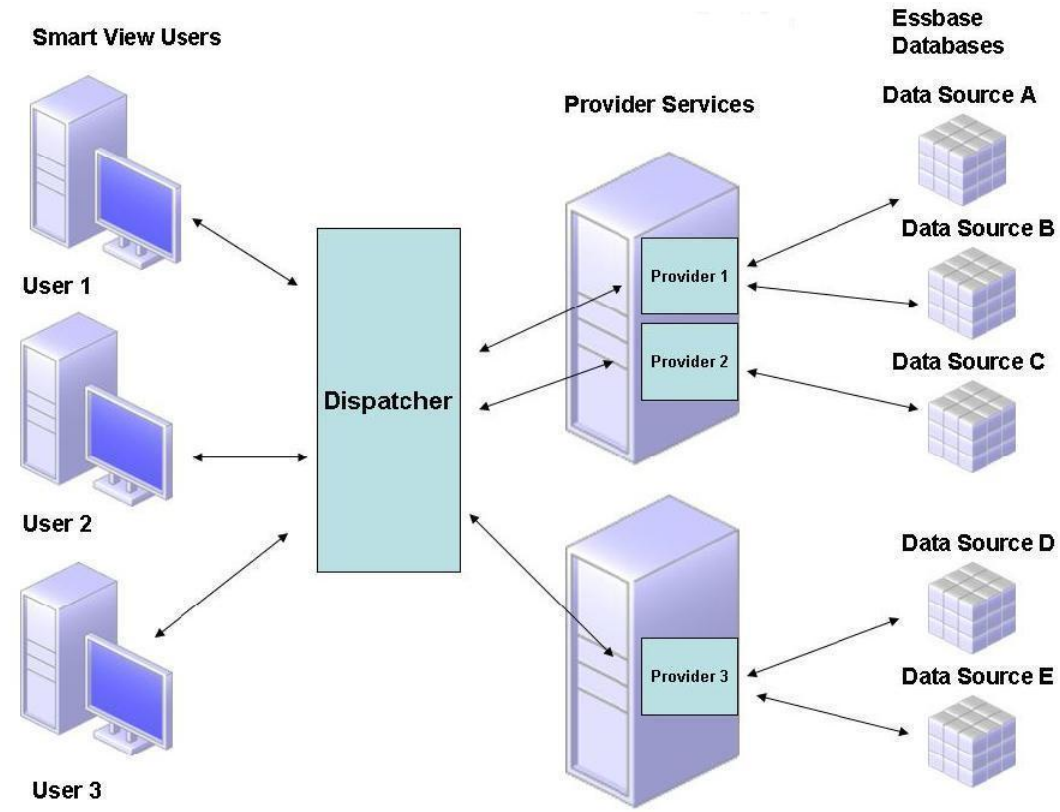
Figure 4 Vertical Provider Services Cluster



Provider Services Connections to Database Clusters

Clustered Oracle Hyperion Provider Services can connect to clustered or stand-alone Oracle Essbase databases. In Figure 5, Provider 1 connects to clustered data sources A and B. Provider 2 connects to one data source C. Provider 3 connects to clustered data sources D and E. A dispatcher distributes Oracle Hyperion Smart View for Office, Fusion Edition client requests among the providers.

Figure 5 Provider Clusters Connecting to Database Clusters



Performance Scorecard Clusters

Before deploying Performance Scorecard to a Java application server cluster, you must complete these steps:

1. Create a local temp folder on the hard disk of each computer in the cluster to contain temporary files; for example, `C:/temp/hpstemp`.

   This local temp folder must exist on all physical machines that join the cluster, and the remaining steps in this procedure must be completed on each machines before deployment. The remainder of this procedures assumes that the directory `c:/temp/hpstemp` exists on all machines.

   Tip: For better performance, create this folder on a high-speed disk.

2. Edit the `oscache.properties` file in the WAR file under `/WEB-INF/classes` of *`HYPERION_HOME`*`/products/PerformanceScorecard/`*`AppServer`*`/InstallableApps/common/webapps/config/HPSWebReports.war` to set this property:

   ```
   cache.path=C:/temp/hpstemp/cache
   ```

3. Using another drive on the local hard disk as *SharedDrive*, map the *HYPERION_HOME*/deployments/webappsconf folder on the machine where Performance Scorecard is installed to this drive shared on the network.

   All machines in the cluster must map the same drive letter (Windows) or the same mount point (UNIX) to this shared drive. This includes the installation machine; that is, it must have a mapping to its own shared drive. This enables all instances in the cluster to share some configuration information and data.

   - The value of parameter HPSConfig in web.xml under the WEB-INF folder of *HYPERION_HOME*/products/PerformanceScorecard/*AppServer*/InstallableApps/common/webapps/config/HPSWebReports.war is set to real:*SharedDrive*/config/HPSConfig.properties.
   - The value of parameter HPSConfig in web.xml under the WEB-INF folder of *HYPERION_HOME*/products/PerformanceScorecard/*AppServer*/InstallableApps/common/webapps/config/HPSAlerter.war is set to real:*SharedDrive*/config/AlerterConfig.properties.

4. In *HYPERION_HOME*/deployments/*Web Server*/HPSWebReports/webappsconfig/config/HPSConfig.properties, set these properties manually to point to *SharedDrive*:

   - hyperion.hps.objectAttachmentsFolder=*SharedDrive*/webappsconf/attachments
   - hyperion.hps.notes.home=C:\\temp\hpstemp\\notes
   - hyperion.hps.main_data_store.castor_mapping_xml=real:*SharedDrive*/config/mapping.xml
   - hyperion.hps.main_data_store.bo_to_jdo_mapping=*SharedDrive*/config/HPSBOToJDOMapping.properties
   - hyperion.hps.main_data_store.dbcp_template=*SharedDrive*/config/dbcp_template.xml

5. In *HYPERION_HOME*/deployments/*Web Server*/HPSWebReports/webappsconfig/config/AlerterConfig.properties, set these properties manually to point to the shared directory:

   - hyperion.hps.main_data_store.castor_mapping_xml=real:*SharedDrive*config/mapping.xml
   - hyperion.hps.main_data_store.bo_to_jdo_mapping=*SharedDrive*/config/HPSBOToJDOMapping.properties
   - hyperion.hps.main_data_store.dbcp_template=*SharedDrive*/config/dbcp_template.xml

For instructions on setting up Oracle Hyperion Performance Scorecard, Fusion Edition clusters on a specific Java application server, see the Java application server documentation.

A Forward Paths For Web Server Routing

In This Appendix

Forward Paths

The following URIs constitute the dynamic content forward paths for which the plug-in must be configured to route requests to the Web application server:

Shared Services

```
/interop
/interop/*
```

EPM Workspace

```
/workspace/administration/*
/workspace/browse/*
/workspace/cdsrpc
/workspace/conf/*
/workspace/dataaccess/*
/workspace/ihtml/*
/workspace/jobmanager/*
/workspace/js/*
/workspace/logon
/workspace/logon/*
/workspace/media/*
/workspace/modules/*
/workspace/personalpages/*
/workspace/portletservlet
/workspace/prefs
/workspace/viewmanager/*
/workspace/wsrp4j/*
/workspace/BPMContext
/workspace/DynamicHelp
/workspace/PDFView
/workspace/ResourceProxy
/workspace/*.jsp
/workspace/*.jsv
```

```
/workspace/*.jsw
/biplus_webservices/*
```

Performance Management Architect

```
/awb
/awb/*
/DataSync
/DataSync/*
```

Calculation Manager

```
/calcmgr
/calcmgr/*
```

In `httpd.conf` add the following proxy requirement for Hyperion Calculation Manager:

```
http://<CalcMgrServer>:<PortNumber>/calcmgr
```

Where `<CalcMgrServer>` is the name of your Calculation Manager server machine and `<PortNumber>` is the number of the Calculation Manager Web application listen port defined in EPM System Configurator (the default port number is 8500).

Administration Services

```
/eas
/eas/*
/easconsole
/easconsole/*
/easdocs
/easdocs/*
/hbrlauncher
/hbrlauncher/*
```

Provider Services

```
/aps
/aps/*
```

Application Builder for .NET

```
/EssbaseObjects
/EssbaseObjects/*
```

Financial Reporting

```
/hr
/hr/*
```

Web Analysis

```
/WebAnalysis/ServerConsole
/WebAnalysis/templates
/WebAnalysis/hfmtemplates
/WebAnalysis/hitemplates
/WebAnalysis/portlettemplates
/WebAnalysis/servlet
/WebAnalysis/modules
/WebAnalysis/resources
/WebAnalysis/processor
/WebAnalysis/DirectoryServlet
/WebAnalysis/config
/WebAnalysis/Config
/WebAnalysis/wa_javadocs/*
/WebAnalysis/*.jsp
/WebAnalysis/*.xml
/WebAnalysis/*.jar
/WebAnalysis/*.exe
```

Planning

```
/HyperionPlanning
/HyperionPlanning/*
```

Performance Scorecard

```
/HPSWebReports
/HPSWebReports/*
/HPSAlerter
/HPSAlerter/*
```

Profitability and Cost Management

```
/profitability
/profitability/*
```

Oracle BI EE

```
/analytics
/analytics/*
```

BI Publisher

```
/xmlpserver
/xmlpserver/*
```

ERP Integrator

```
/aif
/aif/*
```

Static Forward Paths

EPM Workspace

```
/workspace/css
/workspace/docs
/workspace/images
/workspace/img
/workspace/CrystalBall
/workspace/SmartView
/workspace/themes
/workspace/thirdparty
/workspace/wsmedia
/workspace/zeroadmin
```

Web Analysis

```
/WebAnalysis/images
/WebAnalysis/js
/WebAnalysis/resources
/WebAnalysis/themes
```