

Peregrine

AssetCenter Web 4.3

Installation and User's Guide



© Copyright 2004 Peregrine Systems, Inc.

All Rights Reserved.

Information contained in this document is proprietary to Peregrine Systems, Incorporated, and may be used or disclosed only with written permission from Peregrine Systems, Inc. This manual, or any part thereof, may not be reproduced without the prior written permission of Peregrine Systems, Inc. This document refers to numerous products by their trade names. In most, if not all, cases these designations are claimed as Trademarks or Registered Trademarks by their respective companies.

Peregrine Systems® and AssetCenter® are trademarks of Peregrine Systems, Inc. or its subsidiaries.

This document and the related software described in this manual are supplied under license or nondisclosure agreement and may be used or copied only in accordance with the terms of the agreement. The information in this document is subject to change without notice and does not represent a commitment on the part of Peregrine Systems, Inc. Contact Peregrine Systems, Inc., Customer Support to verify the date of the latest version of this document.

The names of companies and individuals used in the sample database and in examples in the manuals are fictitious and are intended to illustrate the use of the software. Any resemblance to actual companies or individuals, whether past or present, is purely coincidental.

If you need technical support for this product, or would like to request documentation for a product for which you are licensed, contact Peregrine Systems, Inc. Customer Support by email at support@peregrine.com.

If you have comments or suggestions about this documentation, contact Peregrine Systems, Inc. Technical Publications by email at doc_comments@peregrine.com.

This edition applies to version 4.3 of the licensed program

AssetCenter

Table of Contents

Foreword	11
Chapter 1. Introduction	17
Operating principles	17
Supported environments	19
Chapter 2. Installation strategies	21
Typical installation or Custom installation?	21
Development environment or production environment?	22
Migration from previous versions	25
Chapter 3. Installing in Windows	29
Using standard application (Tomcat) and Web (Apache) servers	29
Using an alternative application or Web server	29
Typical installation	58
Custom installation	61
Uninstallation	65
Tests	66
Chapter 4. Installing in UNIX	69

Using standard application (Tomcat) and Web (Apache) servers	69
Using an alternative application or Web server	69
Typical installation	86
Custom installation	91
Uninstallation	100
Tests	101
Chapter 5. Load balancing for application servers in Windows	103
Load balancing on application servers	103
Creating multiple instances of Tomcat for Apache	105
Creating multiple instances of Tomcat for IIS	115
Chapter 6. Implementing the servers	129
Starting Tomcat	129
Starting Apache	130
Chapter 7. Configuration	131
Connecting to the administration portal	131
Administration menu	132
Control Panel	132
AssetCenter connector	134
Available languages	135
User rights	136
Defining the AssetCenter Web personalization rights	139
Validating new parameters	140
Modifying a parameter	140
Chapter 8. Utilization	143
AssetCenter demonstration database	143
General interface	144
Default Web applications	146
Example of function	147
Displayed data	151
Customizing AssetCenter Web pages	152
Chapter 9. Troubleshooting	155
Connecting to the AssetCenter database	157
Apache in Windows	157
Accessing the Web server	159
Apache in UNIX	160
IBM HTTP Server	161

Tomcat	161
OAA	165
WebSphere	166
Index	167

List of Figures

1.1. AssetCenter Web - Overview	18
3.1. WebSphere in Windows - Deploying the portal.<Version>.war file	42
3.2. WebSphere in Windows - Selecting the virtual folder	43
3.3. WebSphere in Windows - Selecting the application server	44
3.4. WebSphere in Windows - JMV settings tab	45
3.5. Custom installation in Windows - selecting packages	63
3.6. Custom installation in Windows - CATALINA_OPTS	64
3.7. Custom installation in Windows - Memory settings for Tomcat	65
4.1. WebSphere in Windows - Deploying the portal.<Version>.war file	74
4.2. WebSphere in Unix - Selecting the virtual folder	75
4.3. WebSphere in UNIX - Selecting the application server	75
4.4. WebSphere in UNIX - JMV settings tab	77
4.5. Typical installation in UNIX - Peregrine Portal folder	89
4.6. Typical installation in UNIX - Typical or Custom?	90
4.7. Typical installation in UNIX - Start OAA?	90
4.8. Custom installation in UNIX - Peregrine Portal folder	93
4.9. Custom installation in UNIX - Typical or Custom?	94
4.10. Custom installation in UNIX - Replacing the packages.oaa.ini file	94
4.11. Custom installation in UNIX - Application server to configure	95
4.12. Custom installation in UNIX - Folder of WebSphere application server	95
4.13. Custom installation in UNIX - Application server folder	96
4.14. Custom installation in UNIX - Peregrine Enterprise Portal presentation folder	96

4.15. Custom installation in UNIX - JDK folder used by WebSphere	97
4.16. Custom installation folder in UNIX - JDK folder used by the application server	97
4.17. Custom installation folder in UNIX - IBM HTTPD folder	98
5.1. Load balancing for application servers - architecture	105
7.1. Administration portal - Control Panel	133
8.1. AssetCenter Web - Home page	144
8.2. AssetCenter Web - Searching for an asset	148
8.3. AssetCenter Web - List of assets	150
8.4. AssetCenter Web - Detail of a portfolio item	151
9.1. Troubleshooting - Resolution process	156

List of Tables

1. Guide - Notational conventions	13
2. Guide - Organization	14
1.1. Supported environments	19
2.1. Migration - Customized .jsp files	26
2.2. Migration - Customization performed with the Tailoring Kit	27
3.1. Using an existing Apache or Tomcat server in Windows - Files to copy	30
3.2. IIS - Creating a jakarta virtual directory	35
3.3. IIS - Creating a oaa virtual directory	35
3.4. WebLogic in Windows - Creating a virtual directory	57
3.5. Typical installation in Windows - components	59
3.6. Typical installation in Windows - ports	59
3.7. Custom installation in Windows - ports	62
3.8. Custom installation in Windows - Packages	63
3.9. Installing in Windows - Standard port numbers	67
4.1. Using an existing Apache or Tomcat server in Unix - Files to copy	70
4.2. WebLogic in Unix - Creating a virtual directory	86
4.3. Typical installation in UNIX - packages	87
4.4. Typical installation in UNIX - ports	87
4.5. Typical installation in UNIX - script to execute	89
4.6. Custom installation in UNIX - ports	91
4.7. Custom installation in UNIX - script to execute	93
4.8. Custom installation in UNIX - DLL environment variables	98

4.9. Custom installation folder in UNIX - Root context of the application server	99
4.10. Installing in UNIX - Standard port numbers	101
5.1. Multiple instances of Tomcat for IIS - Creating a virtual directory jakarta	119
5.2. Multiple instances of Tomcat for IIS - Creating a virtual directory oaa	120
7.1. Settings - Languages	135
7.2. Settings - Defining user rights	136
7.3. Settings - List of access rights	138
8.1. Interface - Toolbar icons	145
8.2. Interface - Page icons	146
9.1. Tomcat in UNIX - Location of log files	162
9.2. Tomcat in UNIX - Default ports	162
9.3. OAA - Log files	165

Foreword

PREFACE

About AssetCenter Web

AssetCenter Web enables Web-based access to the AssetCenter database to all users without having to install the AssetCenter client.

Depending on their access rights, a user can perform different operations on the AssetCenter database. Both simple operations, such as browsing assets, and complex operations such as adding or deleting assets are possible.

AssetCenter Web has the following advantages:

- AssetCenter Web generates the pages to be displayed in real time: The pages displayed at any given moment correspond exactly to live data from the AssetCenter.
- AssetCenter Web verifies the rights of users of AssetCenter Web when they access the AssetCenter database.

About the guide

This manual describes in detail how to install, configure and use AssetCenter Web.

Intended audience

The guide is intended for:

- Administrators who will install, configure and maintain AssetCenter Web. To make the most of this guide, the administrator must be fully acquainted with the following aspects:
 - Installing, configuring and administering the Web server you use
 - Installing, configuring and administering the application server you use
 - Administering and using AssetCenter
 - XML language
 - Scripting languages (ECMAScript, JScript or JavaScript), if you use the Tailoring Kit.
 - Administering and using the application servers and Web servers that will be used
- The needs of the AssetCenter Web users

Additional documentation

For further information, you may consult the following documentation:

- The AssetCenter Web **Release Notes**: Late breaking information and known issues are documented here.

Additional applications

This guide does not contain information on the functioning of AssetCenter. This information is included in the AssetCenter documentation.

Warning:

You must first install AssetCenter and create a database for AssetCenter before installing and configuring AssetCenter Web.

Typographic conventions

The following notational conventions are used throughout this manual:

Table 1. Guide - Notational conventions

Convention	Description
Code	Example of code or command
Fixed width characters	DOS command, function parameter or data formatting.
...	Portion of code or command omitted.
Note:	Note for informational purposes.
Additional information	
Important:	Important information for the user.
Be careful...	
Tip:	Tip.
A tip on using...	
Warning:	Important information for the user.
Warning	
Object	Graphical interface object: a menu, a menu item, a tab or a button.

The following conventions are also used:

- Steps to perform in a given order are presented in a numbered list. For example:
 - 1 First step
 - 2 Second step
 - 3 Third and last step
- All figures and tables are numbered according to the chapter in which they are found and the order in which they are presented. For example, the title of the forth table in chapter two will be: **Table 2-4**.

Organization of the guide

The following table presents the structure of the guide:

Table 2. Guide - Organization

The following chapter...	Contains the following information...
Introduction [page 17]	Introduction, installation requirements, and description of different installation types.
Installation strategies [page 21]	Information on the appropriate installation type.
Installing in Windows [page 29]	Detailed instructions for installing AssetCenter Web on a Windows server.
Installing in UNIX [page 69]	Detailed instructions for installing AssetCenter Web on a UNIX server.
Load balancing for application servers in Windows [page 103]	Information on load balancing for optimum system memory management of the application sever.
Implementing the servers [page 129]	Indications on implementing the AssetCenter Web server and making it available to users.
Configuration [page 131]	Explanations on configuring AssetCenter Web via the Web administration interface.
Utilization [page 143]	Explanations on how to display the Web pages to access the AssetCenter database.
Troubleshooting [page 155]	List of common problems with solutions.

If you are an administrator, we recommend reading this guide in its entirety

If you are a user, we recommend reading the chapter [Utilization](#) [page 143].

Technical support

If case of need, you can contact Peregrine Systems technical support.

To contact technical support:

- 1 Start a Web browser.
- 2 Enter the following URL:

<http://support.peregrine.com>

- 3 Log on using your user name and password.
- 4 Follow the instructions given by the site.
Start by searching in the knowledge base.
It contains articles on all Peregrine products.
- 5 If the knowledge base does not provide the adequate response, try looking in the news groups or in the list of available downloads.

Documentation Web site

For the most up-to-date list of documentation for AssetCenter Web, refer to the documentation page on the technical support Web site.

To display the documentation list:

- 1 Start a Web browser.
- 2 Enter the following URL:

```
http://support.peregrine.com
```

- 3 Log on using your user name and password.
- 4 Select either **Documentation**, or **Release Notes** at the top of the page.
- 5 Click the link to AssetCenter Web.
- 6 Click the link 4.3.
- 7 Click the documentation in the language of your choice.

These files in PDF format can be read using Adobe Acrobat Reader.

This can be downloaded using the link at the Peregrine technical support site or else from Adobe at the following address:

```
http://www.adobe.com
```

Important:

The Release Notes are updated regularly.

Make sure you have the most up to date version.

Training

Peregrine provides a complete range of education services: Classroom-based training, Web training, virtual classes.

For further information, you can connect to the following address:

```
http://www.peregrine.com/education
```

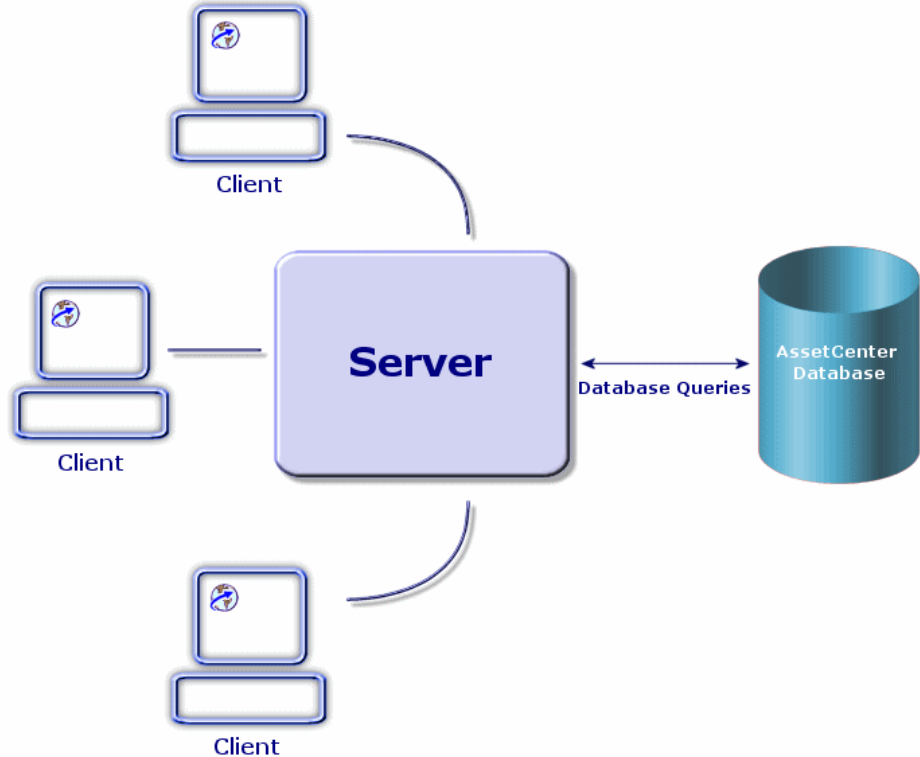

1 Introduction

CHAPTER

Operating principles

AssetCenter Web is a set of Web applications that use the Open Application Architecture technology. This technology was designed in order to:

- Offer a wide range of services to all members of an organization.
- Offer access to these services on a collective or on an individual basis.
- Help an enterprise's employees rapidly accomplish their predefined tasks.

Figure 1.1. AssetCenter Web - Overview

The standard AssetCenter Web application is comprised of:

- The OAA (Open Application Architecture) with the AssetCenter Web pages.
- SDK Java2 Standard Edition, which includes the Java run-time environment.
- The Apache Web server.
- Tomcat Web container.

In addition to these components, you must install an AssetCenter client on the AssetCenter Web server and use this client to declare the connection to the AssetCenter database accessed via AssetCenter Web.

Supported environments

The following table lists what is required for AssetCenter Web to work:

Table 1.1. Supported environments

Component	In Windows	In UNIX
Java Runtime Environment	Java 2 SDK Standard Edition (included on the AssetCenter Web installation CD-ROM)	Java 2 SDK Standard Edition (included on the AssetCenter Web installation CD-ROM)
Application server	One of the following servers: <ul style="list-style-type: none"> • Tomcat (included on the AssetCenter Web installation CD-ROM) • WebSphere Application Server • WebLogic 	One of the following servers: <ul style="list-style-type: none"> • Tomcat (included on the AssetCenter Web installation CD-ROM) • WebSphere Application Server • WebLogic
Database	AssetCenter	AssetCenter
Operating system	Windows	<ul style="list-style-type: none"> • AIX • Red Hat Linux • Solaris
Web server	One of the following servers: <ul style="list-style-type: none"> • Apache (included on the AssetCenter Web installation CD-ROM) • Microsoft IIS Server • IBM HTTP Server 	One of the following servers: <ul style="list-style-type: none"> • Apache (included on the AssetCenter Web installation CD-ROM) • IBM HTTP Server
CPU	Pentium II 400 MHz or better	<ul style="list-style-type: none"> • AIX: POWER 3, 375 MHz or better • Red Hat Linux: Pentium II 400 MHz or better • Solaris: Ultra SPARC II 300 MHz or better
RAM	512 MB or better	512 MB or better
Disk space	100 MB for AssetCenter Web	100 MB for AssetCenter Web



Tip:

This table does not include version numbers. For more specific information, please refer to the Compatibility Matrix.

→ <http://support.peregrine.com>



Note:

The AssetCenter API must be installed on the AssetCenter Web server. The AssetCenter database, however, may be installed on a different server.

2 | Installation strategies

CHAPTER

Typical installation or Custom installation?

The AssetCenter Web installation program offers several installation types:

- **Typical:** Automatically installs the standard set of AssetCenter Web components on the same computer (with Apache as application server and Tomcat as Web server).
- **Custom:** Enables you to select the specific components to be installed.

The enables you to, for example:

- Use alternatives to Apache and/or Tomcat.
 - [Supported environments](#) [page 19].
- To install several instances of Tomcat.
 - [Load balancing for application servers in Windows](#) [page 103].

 **Warning:**

Only use the custom installation method if you know exactly what you are doing. Many parameters need to be configured manually.

Development environment or production environment?

Determining the environment for which you are installing AssetCenter Web

You may install AssetCenter Web for different intended environments:

- Development environment
- Production environment

Development environment

This environment corresponds to a utilization of AssetCenter Web for learning, testing and customizing purposes before actually using it in a production role.

In this environment, you install all the components required for AssetCenter Web to work independently on the computer.

You have the choice between two installation types: Typical or Custom.

Production environment

This environment corresponds to a utilization of AssetCenter Web optimized for performance and including the customizations made in a development environment.

In this environment, you install the applications required for AssetCenter Web on several servers in order to optimize performance.

You have the choice between two installation types: Typical or Custom.

Installing for a development environment

To install AssetCenter Web with the Typical method for a development environment:

- 1 Assemble the required hardware and software.
- 2 Make sure the AssetCenter database can be reached on a server.
- 3 Install the AssetCenter client and the AssetCenter API on the development computer.

- 4 Declare the connection to the AssetCenter database using the AssetCenter client.
- 5 Uninstall AssetCenter Web if a previous version already exists.
→ [Migration from previous versions](#) [page 25]
- 6 Start the AssetCenter Web installation program.
Select the **Typical** installation type.
In Windows: → [Typical installation](#) [page 58].
In UNIX: → [Typical installation](#) [page 86].
- 7 Create the AssetCenter Web users in the AssetCenter database and give them their appropriate user profiles.
→ [User rights](#) [page 136]

To install AssetCenter Web with the Custom method for a development environment:

- 1 Assemble the required hardware and software.
- 2 Make sure the AssetCenter database can be reached on a server.
- 3 Install the AssetCenter client and the AssetCenter API on the development computer.
- 4 Declare the connection to the AssetCenter database using the AssetCenter client.
- 5 Install and configure the alternative Web server and/or application server on the development server.
In Windows: → [Using an alternative application or Web server](#) [page 29].
In UNIX: → [Using an alternative application or Web server](#) [page 69].
- 6 Uninstall AssetCenter Web if a previous version already exists.
→ [Migration from previous versions](#) [page 25]
- 7 Start the AssetCenter Web installation program.
Select the **Custom** installation type.
In Windows: → [Typical installation](#) [page 58].
In UNIX: → [Typical installation](#) [page 86].
- 8 Create the AssetCenter Web users in the AssetCenter database and give them their appropriate user profiles.
→ [User rights](#) [page 136]

Installing for a production environment

To install AssetCenter Web for a typical production environment:

- 1 Assemble the required hardware and software.
- 2 Make sure the AssetCenter database can be reached on a server.
- 3 Install the AssetCenter client and the AssetCenter API on the development computer.
- 4 Declare the connection to the AssetCenter database using the AssetCenter client.
- 5 Uninstall AssetCenter Web if a previous version already exists.
→ [Migration from previous versions](#) [page 25]
- 6 Start the AssetCenter Web installation program.
Select the **Typical** installation type.
In Windows: → [Typical installation](#) [page 58].
In UNIX: → [Typical installation](#) [page 86].
- 7 Configure the multiple instances of Tomcat to balance the load on the Apache Web server.
→ [Load balancing for application servers in Windows](#) [page 103]
- 8 Create the AssetCenter Web users in the AssetCenter database and give them their appropriate user profiles.
→ [User rights](#) [page 136]

To install AssetCenter Web for a custom production environment:

- 1 Assemble the required hardware and software.
- 2 Make sure the AssetCenter database can be reached on a server.
- 3 Install the AssetCenter client and the AssetCenter API on the development computer.
- 4 Declare the connection to the AssetCenter database using the AssetCenter client.
- 5 Install and configure the alternative Web server and/or application server on different servers.
In Windows: → [Using an alternative application or Web server](#) [page 29].
In UNIX: → [Using an alternative application or Web server](#) [page 69].
- 6 Uninstall AssetCenter Web if a previous version already exists.

- Migration from previous versions [page 25]
- 7 Start the AssetCenter Web installation program.
Select the **Typical** installation type.
In Windows: → Typical installation [page 58].
In UNIX: → Typical installation [page 86].
- 8 Configure the multiple instances of Tomcat to balance the load on the Apache Web server.
→ Load balancing for application servers in Windows [page 103]
- 9 Create the AssetCenter Web users in the AssetCenter database and give them their appropriate user profiles.
→ User rights [page 136]

Migration from previous versions



Note:

This documentation does not deal with AssetCenter migration issues. Please refer to the AssetCenter documentation.

To migrate from a previous version of AssetCenter Web:

- 1 Make a detailed description of all customizations of the previous version of AssetCenter Web.
 - 2 Review the customizations of the previous version of AssetCenter Web and determine which customizations need to be recreated in version 4.3.
 - 3 Install AssetCenter Web version 4.3 on a new server.
 - 4 Apply any customizations to AssetCenter Web you want to propagate manually.
-



Warning:

You cannot directly migrate customizations implemented in previous versions of AssetCenter Web.

Instead, you must recreate your changes using the new features and methods available in version 4.3.

Customized .jsp files

In previous versions of AssetCenter Web, customers had to directly modify **.jsp** files in order to add or remove certain functionality.

The following table describes how to recreate some of the more common JSP file modifications with version 4.3 of AssetCenter Web:

Table 2.1. Migration - Customized .jsp files

Modification with the previous version	Method to use with version 4.3
Remove the user self-registration option from login page	<p data-bbox="578 541 1219 618">To enable or disable the user registration option from the Administration Settings page:</p> <ol style="list-style-type: none"> <li data-bbox="578 638 1219 708">1 Access the administration portal with your Web browser → Connecting to the administration portal [page 131] <li data-bbox="578 715 953 741">2 Display the Administration tab <li data-bbox="578 748 968 774">3 Select the Admin/ Settings menu <li data-bbox="578 781 893 807">4 Display the Common tab. <li data-bbox="578 814 1093 861">5 Modify the Enable User Registration setting.
Remove the change password option from the login page	

Personalized pages

Many aspects of AssetCenter Web pages can now be modified directly from a Web browser. The customization tool is called DocExplorer.

If you personalized pages in a previous version of AssetCenter Web, you must recreate your personalized pages using DocExplorer (adding or removing fields or links, for example).

→ [Defining the AssetCenter Web personalization rights \[page 139\]](#) et → [Customizing AssetCenter Web pages \[page 152\]](#)

Customized skins, stylesheets, and themes

All interface images and stylesheets have been combined into themes. Users can no longer select separate skins and stylesheets.

The new themes include the following elements:

- Skins (which themselves are composed of image files, frame definitions, and layer files)
- Cascading stylesheet definitions
- XSL templates

Although you may copy over older custom themes to AssetCenter Web 4.3, you may experience rendering errors due to the new images, CSS definitions, frame definitions, and layers.

It is recommended that you recreate any custom themes using the AssetCenter Web 4.3 version of the classic theme as your template.

Alternate login pages and authentication methods

If you used a custom login page or an alternate authentication method in a previous version, you can re-use or recreate these customizations using the updated instructions.

Customizations made with a previous tailoring kit

Many customizations that required a tailoring kit in previous versions can now be made directly from the AssetCenter Web Web interface.

The following table describes how to recreate some of the more common tailoring kit changes:

Table 2.2. Migration - Customization performed with the Tailoring Kit

Modification performed with the Tailoring Kit	New method
Add or remove a field or link from a page	Add or remove a field directly from the AssetCenter Web interface using DocExplorer (🔍 icon) → Customizing AssetCenter Web pages [page 152] .
Added a new language or locale to the AssetCenter Web interface	Create and edit language strings files directly. You may also purchase officially supported language packs from Peregrine Systems.

Modify the **Common, Portal or Peregrine Studio** packages

These packages are no longer available for tailoring.

However most common interface settings can now be customized from the Administration Settings page.

- 1 Display the **Administration** tab
- 2 Select the **Admin/ Settings** menu
- 3 Display the **Common** and **Portal** tabs

Modify the ECMA server-side scripts

Review new functionality and determine if you still need the customized scripts and schemas.

If you do need the customizations, you will to recreate them in the current version of the AssetCenter Web version 4.3 tailoring kit.

Upgrading a previous version of AssetCenter Web

To upgrade a previous version of AssetCenter Web to version 4.3:

- 1 Uninstall the previous version of AssetCenter Web in full (**Start/ Settings/ Control Panel/ Add or Remove Programs** in Windows XP).
- 2 Execute the AssetCenter Web installation program.
 In Windows: → [Typical installation](#) [page 58] or →
 In UNIX: → [Typical installation](#) [page 58] or →

3 | Installing in Windows

CHAPTER

Chapter [Installation strategies](#) [page 21] enabled you to define your administration strategy.

This chapter explains how to implement this strategy in the Windows environment.

Using standard application (Tomcat) and Web (Apache) servers

If you wish to use Tomcat and Apache, and they are not yet installed on the AssetCenter Web server(s), you can read section [Typical installation](#) [page 58] directly.

Using an alternative application or Web server

AssetCenter Web requires a Web server and a Java-enabled application server to run.

For the list of supported servers: → [Supported environments](#) [page 19].

By default, the installation program installs Tomcat and Apache.

If you already use other combinations of servers, you must install and configure these **before** executing the AssetCenter Web installation program.

If you already have the Apache and Tomcat servers

If Apache and Tomcat are already installed, you can configure AssetCenter Web so that it uses these external applications instead of its default applications by simply installing the **AssetCenter Web** package using the **Custom** installation.

To install and configure AssetCenter Web so that it uses existing Tomcat and Apache servers:

- 1 Stop active instances of Tomcat and Apache.
- 2 Start the AssetCenter Web installation program and select the **Custom** installation type.

In the **Select Components** window, select the **AssetCenter Web** option only.

In the **Select Application Server Type** window, select **Tomcat**.

In the **Select Web Server** window, select **Apache Web Server**.

→ [Custom installation](#) [page 61].

- 3 Copy the following files (located in the **acw\SupportFiles** folder of the AssetCenter installation CD):

Table 3.1. Using an existing Apache or Tomcat server in Windows - Files to copy

Source file	Destination
acw\SupportFiles\tomcat\conf\mod_jk.conf	Conf sub-folder of Tomcat. Example: C:\Program Files\Apache Group\Tomcat 4.1\conf.
acw\SupportFiles\tomcat\conf\workers.properties	Conf sub-folder of Tomcat. Example: C:\Program Files\Apache Group\Tomcat 4.1\conf.
acw\SupportFiles\Apache\modules\mod_jk.dll	Modules sub-folder of the Apache installation folder. Example: C:\Program Files\Apache Group\Apache 2\modules.



Note:

The **mod_jk.conf** file provided with AssetCenter Web is compatible with the versions of Apache and Tomcat that are indicated in the AssetCenter compatibility matrix.

If you use different versions, refer to the following site to download the compatible version:

```
jakarta.apache.org/builds/jakarta-tomcat-connectors/jk/doc
```

- 4 Using a text editor, open the files **mod_jk.conf** and **workers.properties**. (These files are located in the **conf** folder of the Tomcat installation folder.)

Make sure the access paths of Tomcat point to the Tomcat installation folder.

The places to populate are given by the string **PRGNTOMCATDIR**, which you simply need to search.

Do the same for the JDK access paths given by the **PRGNJAVAHOME** string.

- 5 Using a text editor, open the **httpd.conf** file. (This file is located in the **conf** folder of the Apache installation folder).

- Add the access path of Tomcat to the **Global Environment** section, using the following **include** clause:

```
### Section 1: Global Environment
include "<Full path of the Tomcat installation folder>/conf/mod_
jk.conf"
```

- Search the following string:

```
DirectoryIndex index.html index.html.var
```

Replace it with:

```
DirectoryIndex index.html login.jsp
```

- Search the following string:

```
Alias /oaa ""
```

Replace it with:

```
Alias /oaa "<Full path of the Tomcat installation folder>\webapp
s\oaa"
```

- 6 Start Apache and Tomcat.

- 7 Start a Web browser and enter the address of the AssetCenter Web login page.
⇒ [Connecting to AssetCenter Web \[page 144\]](#).
Make sure you can connect.

Tomcat with IIS

This section explains how to configure Tomcat so that it uses a single instance of Java Virtual Machine (JVM).

If you wish to use multiple instances of JVM: ⇒ [Load balancing for application servers in Windows \[page 103\]](#)

Global process

- 1 Install AssetCenter Web [\[page 32\]](#)
 - 2 Configuring the ISAPI plugin for IIS [\[page 33\]](#)
 - 3 Configuring IIS to use isapi_redirector2.dll as an ISAPI filter [\[page 33\]](#)
 - 4 Creating and configuring a jakarta virtual directory in IIS [\[page 34\]](#)
 - 5 Creating and configuring a virtual directory oaa in IIS [\[page 35\]](#)
 - 6 Start the Web server [\[page 36\]](#)
 - 7 Improving performance and changing the ports used by Tomcat [\[page 37\]](#)
-



Note:

These instructions configure Tomcat so that it uses one single instance of Java Virtual Machine (JVM).

To use multiple instances of JVM: ⇒ [Load balancing for application servers in Windows \[page 103\]](#).

Install AssetCenter Web

To install AssetCenter Web:

- 1 Execute the AssetCenter Web installation program.
- 2 Select the **Custom** type.
- 3 In the **Select Components** window, select all the packages except **Apache**.
- 4 In the **Select Web Server** window, select **Other**.

→ Custom installation [page 61]

Configuring the ISAPI plugin for IIS

The ISAPI plugin for IIS establishes a connection between Tomcat and the IIS Web server.

Before configuring IIS to use this connector, you must update the registry file entry for the connector to ensure that it has the proper paths listed for the Tomcat application server.

The AssetCenter Web installation program automatically places a copy of the ISAPI plugin for IIS in the folder **c:\Program**

Files\Peregrine\Common\Tomcat4\bin:

To configure the ISAPI plugin for ISS for your intranet environment.

- 1 Open the **jk2.reg** file (located by default in the folder **C:\Program Files\Peregrine\Common\Tomcat4\conf**) in a text editor.
- 2 Make sure the **serverRoot** and **workersFile** values correctly reference the Tomcat installation folder. By default, these values are:

```
"ServerRoot"="C:\\Program Files\\Peregrine\\Common\\Tomcat4"
"workersFile"="C:\\Program Files\\Peregrine\\Common\\Tomcat4\\conf\\
workers2.properties"
```

- 3 Save and close the **jk2.reg** file.
- 4 In the Windows explorer, double-click the **jk2.reg** file.

This modifies the Registry.

Configuring IIS to use isapi_redirector2.dll as an ISAPI filter

To establish a connection between Tomcat and IIS, you must configure isapi_redirector2.dll :

- 1 Start the Windows Internet Services management console (**Start/ Settings/ Control Panel/ Administrative Tools/ Internet Services (IIS)** menu in Windows XP).
- 2 Right-click the **Default Web Site** node and select the **Properties** menu.
- 3 Select the **ISAPI Filters** tab.
- 4 Click **Add**.
- 5 Set the **Filter Name** field to **jakarta**.

The filter name must match the name defined in the **jk2.reg** file. (located by default in the folder **C:\Program Files\Peregrine\Common\Tomcat4\conf**).

By default, this name is **jakarta**.

This line that gives this name is:

```
"extensionUri"="/jakarta/isapi_redirector2.dll"
```

- 6 Populate the **Executable** field with the full path of the **isapi_redirector2.dll** file.
By default, this file is located in **C:\Program Files\Peregrine\Common\Tomcat4\bin**.
- 7 Click **OK**.
- 8 Restart the **IIS Admin** and **Word Wide Web Publishing** services.
- 9 Restart the **PeregrineTomcat** service.
- 10 Start the Windows Internet Services management console (**Start/ Settings/ Control Panel/ Administrative Tools/ Internet Services (IIS)** in Windows XP).
- 11 Right-click the **Default Web Site** node and select the **Properties** menu.
- 12 Select the **ISAPI Filters** tab.
- 13 Verify that the service is running (green status arrow).

Creating and configuring a jakarta virtual directory in IIS

The ISAPI plugin for IIS requires a specific virtual directory in order to run. Use the following guidelines to create the virtual directory on the Default Web Site.

For specific instructions about configuring IIS, refer to your Windows Help.

To configure a jakarta virtual directory in IIS:

- 1 Start the Windows Internet Services management console (**Start/ Settings/ Control Panel/ Administrative Tools/ Internet Services (IIS)** menu in Windows XP).
- 2 Right-click the **Default Web Site** node and select the **New/ Virtual directory** menu.
- 3 Create the virtual directory using the following parameters:

Table 3.2. IIS - Creating a jakarta virtual directory

Specification	Value
Virtual directory alias	jakarta
Web site content directory	<Tomcat installation folder>\bin The default installation folder of Tomcat is: C:\Program Files\Peregrine\Common\Tomcat4
Access authorizations	Read, Execute scripts, Execute

Creating and configuring a virtual directory oaa in IIS

To execute AssetCenter Web with IIS, you must create a virtual directory that maps the Tomcat installation folder.

For specific instructions about configuring IIS, refer to your Windows Help.

To configure a oaa virtual directory in IIS:

- 1 Start the Windows Internet Services management console (**Start/ Settings/ Control Panel/ Administrative Tools/ Internet Services (IIS)** menu in Windows XP).
- 2 Right-click the **Default Web Site** node and select the **New/ Virtual directory** menu.
- 3 Create the virtual directory using the following parameters:

Table 3.3. IIS - Creating a oaa virtual directory

Specification	Value
Virtual directory alias	oaa
Web site content directory	<Tomcat installation folder>\webapps\oaa The default installation folder of Tomcat is: C:\Program Files\Peregrine\Common\Tomcat4
Access authorizations	Read, Execute scripts

<Tomcat installation folder>\webapps\oaa: We recommend using the name **oaa**.

If you chose another name, you must replace **oaa** everywhere else it is referenced:

- **<Tomcat installation folder>\webapps\oaa** folder.
- [uri] mappings in the **workers2.properties** file.
- **oaa** entries in the **mod_jk.conf** file.
- **<Context** path and **docBase** attributes of the **server.xml** file.

 **Important:**

The other virtual directory name must appear in the access URL of AssetCenter Web.

Example: `http://<Server name>/<Name of the virtual directory>/login.jsp`

In order for the login page to display correctly when you enter the URL `http://<Server name>:<port>/oaa:`

- 1 Start the Windows Internet Services management console (**Start/ Settings/ Control Panel/ Administrative Tools/ Internet Services (IIS)** in Windows XP).
- 2 Unfold the **Default Web Site** node.
- 3 Right-click the **oaa** virtual folder and select **Properties**.
- 4 Select the **Documents** tab.
- 5 Verify that **Enable Default Document** is selected.
- 6 Click **Add**.
- 7 Type **login.htm**.
- 8 Click **OK**.
- 9 Highlight **login.htm** and using the up and down arrows, move login.htm to the top of the file list.
- 10 Click **OK**.

Start the Web server

- 1 Start the Windows Internet Services management console (**Start/ Settings/ Control Panel/ Administrative Tools/ Internet Services (IIS)** in Windows XP).
- 2 Unfold the **Default Web Site** node.

- 3 Right-click the **Default Web Site** node.
- 4 Select the **Start** menu.
- 5 Start a Web browser and enter the address of the AssetCenter Web login page.
→ [Connecting to AssetCenter Web](#) [page 144].
Make sure you can connect.

Improving performance and changing the ports used by Tomcat

In general, the default configuration of Tomcat is sufficient.

However, if you are experiencing performance problems or communications port conflicts, you may need to edit the Tomcat **server.xml** file.

Performance settings

The **server.xml** file enables you to determine how Tomcat processes the AssetCenter Web files.

If you are experiencing performance problems, you can change the **<Context** setting for AssetCenter Web to disable page reloading.



Tip:

Make a back up copy of the **server.xml** file before editing.

To edit the server.xml performance settings:

- 1 Open the **server.xml** file in a text editor.
By default, this file is located in the folder **C:\Program Files\Peregrine\Common\Tomcat4\conf**.
- 2 Create a **<Context** element entry from Tomcat to the installation folder to establish a point of reference for **docBase**.
Add the following entry just before the **<Context path="/examples"** entry.
Example:

```
<Context path="/oaa"
docBase="<Absolute path of the first or master Tomcat instance>/web
apps/oaa">
crossContext="false"
debug="0"
reloadable="false">
</Context>
```

- Setting the **reloadable** value to **false** results in faster JSP page processing.
- Restart Tomcat.

Communication port settings

The **server.xml** file enables you to determine how Tomcat processes the AssetCenter Web files.

If the server that hosts AssetCenter Web already uses communications ports **8005** and **8009**, you will have a port conflict if you install Tomcat with the default settings.

To avoid a port conflict, you must edit the **server.xml** file to change the communications ports used by Tomcat.



Tip:

Make a back up copy of the **server.xml** file before editing.

To modify the server.xml communications port settings:

- Open the **server.xml** file in a text editor.
By default, this file is located in the folder **C:\Program Files\Peregrine\Common\Tomcat4\conf**.
- Update the port number attribute of the `<Server` element to a free communications port.



Note:

By default, Tomcat uses port 8005 for shutdown requests.

Example:

```
<Server port="8005" shutdown="SHUTDOWN" debug="0">
```

- Update the **port** attribute of the Coyote Connector `<Connector` element to a free communications port.

It is possible to recognize the correct `<Connector` element because it contains the following line:

```
protocolHandlerClassName="org.apache.jk.server.JkCoyoteHandler
```



Note:

By default, Tomcat uses port 8009 for the Coyote connector.

Example:

```
<Connector className="org.apache.coyote.tomcat4.CoyoteConnector" port="8009"
minProcessors="5" maxProcessors="75" enableLookups="true" redirectPort="8443"
acceptCount="10" debug="0" connectionTimeout="20000"
useURIVValidationHack="false"
protocolHandlerClassName="org.apache.jk.server.JkCoyoteHandler"/>
```

- 4 Save the **server.xml** file.
- 5 Restart Tomcat so that the new parameters are taken into account.

WebSphere 4.0

This section describes the procedure to configure WebSphere 4.0 to be used as a substitute for Apache and Tomcat.



Note:

The AssetCenter Web installer creates duplicate alias entries in the IBM HTTP Server when you install more than one Peregrine OAA Platform application on WebSphere.

Duplicate entries can also occur if you reinstall AssetCenter Web or install another Peregrine OAA Platform application on a system that formerly had AssetCenter Web installed on it.

Remove any duplicate alias entries from the IBM HTTP Server **httpd.conf** file.

To configure WebSphere 4.0:

- 1 Installing WebSphere 4.0 [page 40]
- 2 Deploying the portal.<Version>.war file to WebSphere [page 41]
- 3 Setting the heap size parameter [page 44]
- 4 Configuring a virtual directory for IBM HTTP Server [page 46]
- 5 Execute the AssetCenter Web installation program [page 46]
- 6 Regenerating the plug-in configuration [page 46]

If you plan on setting up WebSphere Translation Server: → WebSphere Translation Server [page 81].

Installing WebSphere 4.0



Tip:

To check which version of WebSphere and the fix pack are installed, go to the **WebSphere\AppServer\logs\default_server_Stdout.log** file.



Note:

WebSphere includes IBM HTTP Server.

To install WebSphere 4.0:

- 1 Make sure IIS is stopped: The **World Wide Web Publishing** service must be stopped and configured to be started automatically.
- 2 Give a user the following rights (**Local security policy/ Local Policies/ User rights assignment** menu):
 - Act as part of the operating system
 - Create a token object
 - Increase quotas
 - Replace a process level token
- 3 Restart the computer.
- 4 Start the DB2 7.2 server installation program. Respect the following points:
 - Select **DB2 Enterprise Edition** and **DB2 Administration Client**
 - Select the typical installation
 - Select the user you have created
 - Do not install OLAP
- 5 Stop all the **DB2*** services, making sure you stop the **DB2 - DB2** service last.
- 6 Execute **C:\Program Files\SQLLIB\java12\usejdbc2.bat**.
- 7 Restart the computer.
- 8 Install WebSphere Application Server 4, selecting the typical installation.
- 9 Restart the computer.

- 10 Remove all the unused entries from the Windows **Start/ Programs/ Startup** menu.
- 11 Stop all **IBM*** services.
- 12 Execute **WebSphere\fixpack5\install.bat** from the WebSphere installation CD-ROM.
Confirm at each step, except for updating iPlanet and d'Apache, which you will refuse.
- 13 Restart the computer.
- 14 Start the **IBM WS AdminServer 4.0** service and set its startup type to **Automatic**.
- 15 Start WebSphere Administrator's Console.



Tip:

It is normal that the WebSphere Administrator's Console takes some time to start up.

- 16 Go to the node corresponding to your computer.
- 17 Right click.
- 18 Select **regen plug-in configuration** from the menu.
- 19 If necessary, right-click the node corresponding to the server and then start the node.
- 20 Make sure the installation is successful by pointing an Internet browser to **http://localhost**.

Deploying the portal.<Version>.war file to WebSphere

The **portal.<Version>.war** file creates the folder structure necessary to deploy AssetCenter Web with the application server.

Once this file is deployed, it is possible to execute the AssetCenter Web installation program.

To deploy the portal.<Version>.war file to WebSphere:

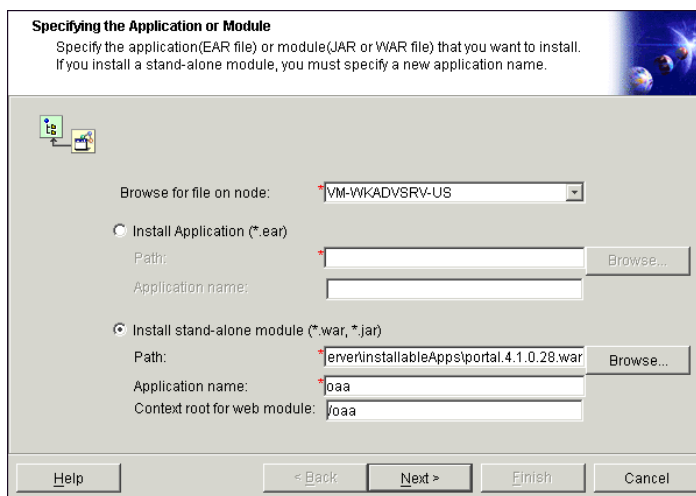
- 1 Copy **<AssetCenter installation CD-ROM>\acw\packages\portal.<most recent version>.war** to the folder **C:\WebSphere\AppServer\installableApps**.
- 2 Verify that the WebSphere Admin Server has been started.

- 3 Start the WebSphere Advanced Administrator's Console (**Start/ Programs/ IBM WebSphere/ Application Server/ Administrator's Console** menu in Windows XP).
- 4 On the menu at the left side of the console, right-click **Enterprise Applications** and select **Install Enterprise Application**.
- 5 In the window that is displayed:
 - Select **Install stand-alone module**.
 - **Path** field: Browse to the folder containing the **portal<most recent version>.war** file.
 - **Application Name** field: Type **oaa**.
 - **Context Root** field: Type the name of the virtual Web server folder you wish to associate with AssetCenter Web.
Example: /oaa.

 **Important:**

You must create a virtual directory matching the path you enter here.

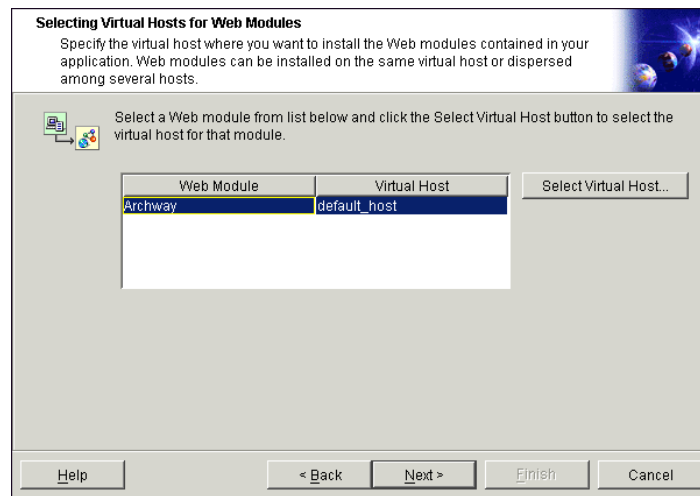
Figure 3.1. WebSphere in Windows - Deploying the portal.<Version>.war file



- 6 Click **Next** without modifying the following screens:
 - Mapping EJB Run As Roles to Users

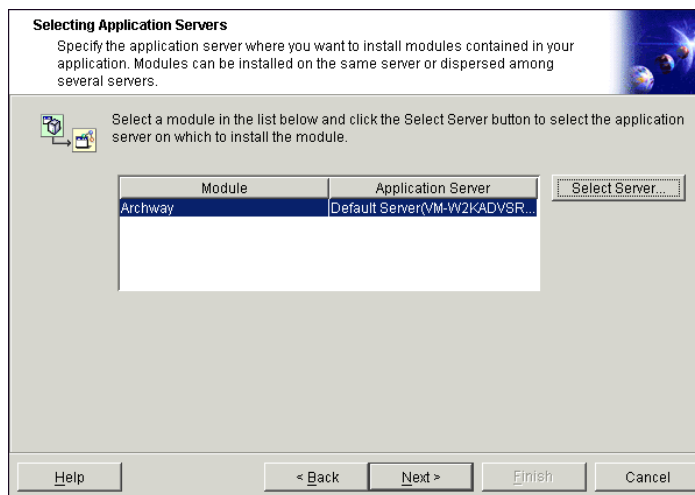
- Binding Enterprise Beans to JNDI Names
 - Mapping EJB References to Enterprise Beans
 - Mapping Resource References to Resources
 - Specifying the Default Datasource
 - Specifying Data Sources for Individual CMP Beans
- 7 In the **Selecting Virtual Hosts for Web Modules** window, select the instance of the WebSphere server to use, and then click **Next**.

Figure 3.2. WebSphere in Windows - Selecting the virtual folder



- 8 In the **Selecting Application Servers** folder, select the instance of the WebSphere server to use, and then click **Next**.

Figure 3.3. WebSphere in Windows - Selecting the application server



- 9 Click **Finish**.
- 10 Go to **nodes/<Your computer>/Application servers**.
In the **JVM** tab, specify **60** and **225**.
- 11 Close all windows.
- 12 Restart the computer.

Setting the heap size parameter

You can configure how much memory is made available for your application server.

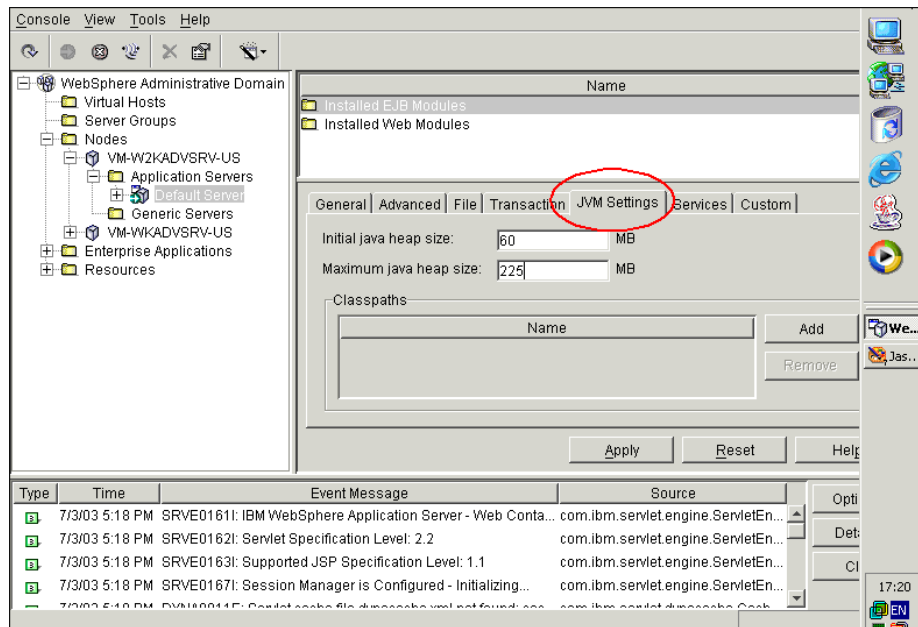
The following instructions assume you are only using one WebSphere instance. You will need to adjust the **heap size** parameter accordingly if you are load balancing across several WebSphere instances.

To set the heap size parameter:

- 1 Verify that the WebSphere Admin Server has been started.
- 2 Start the WebSphere Advanced Administrator's Console (**Start/ Programs/ IBM WebSphere/ Application Server/ Administrator's Console** menu in Windows XP).

- 3 Select **Nodes/ <System Name>/ Application Servers/ <Application server name>**.
- 4 Click the **JMV settings** tab:

Figure 3.4. WebSphere in Windows - JMV settings tab



Populate the following fields:

- **Initial java heap size** field: Type **60**.
- **Maximum java heap size** field: Type the value you want for heap memory.

This setting should be at least 225 MB, but not more than 512 MB.

 **Note:**

Make sure that the setting for **maximum heap size** is less than the free RAM available to the application server(s).

Exceeding the amount of available RAM causes the JVM processes to swap to disk, reducing overall performance. A setting of 256 MB should be sufficient for most systems.

Execute the AssetCenter Web installation program

Execute the installation program and select the **Custom** installation option.

→ [Custom installation](#) [page 61].

Configuring a virtual directory for IBM HTTP Server

You must configure a virtual directory for AssetCenter Web in your Web server. The following instructions assume that you are using IBM HTTP Server as the Web server (WebSphere's default Web server).

See your Web server documentation to determine how to create a virtual directory if you are using another Web server.

To configure a virtual directory for IBM HTTP Server:

- 1 Stop IBM HTTP Server.
- 2 Open the **httpd.conf** file in a text editor.
By default, this file is located in the **C:\IBM HTTP Server\conf** folder.
- 3 Add the following line to the end of the file:

```
Alias /oaa/ "C:/WebSphere/AppServer/installedApps/oaa.ear/portal.<I
nstalled version of .war file>.war/"
```

Important:

The name of the virtual folder must match the value of the **Context Root** field.

→ [Deploying the portal.<Version>.war file to WebSphere](#) [page 41].

- 4 Save the file.
- 5 Start IBM HTTP Server.
- 6 Copy the files **jsse.jar** and **sujce_provider.jar** of jre Sun 1.4.1_03 to the folder **C:\WebSphere\AppServer\java\jre\lib\ext**.

Regenerating the plug-in configuration

To regenerate the plug-in configuration

- 1 Start the WebSphere Advanced Administrator's Console (**Start/ Programs/ IBM WebSphere/ Application Server/ Administrator's Console** menu in Windows XP).

- 2 Select **Nodes/ <System Name>/ Application Servers/ <Application server name>**.
- 3 Right-click **<System Name>**, and then select **Regen Webserver Plugin**.
Copy the following line from the **<settings>** section of the **<Location of the application server>\WEB-INF\default\archway.xml** file to the **<settings>** section of the **<Location of the application server>\WEB-INF\local.xml** file:


```
<SSLProvider>com.ibm.jsse.JSSEProvider</SSLProvider>
<HTTPSHandlerPkg>com.ibm.net.ssl.internal.www.protocol</HTTPSHandlerPkg>
<CryptoProvider>com.ibm.crypto.provider.IBMJCE</CryptoProvider>
```
- 4 Restart your application server.
- 5 Make sure the installation is successful by pointing an Internet browser to **http://localhost**.

WebSphere 5.0

This section describes the procedure to follow to configure WebSphere 5.0 to replace Apache and Tomcat.



Note:

The AssetCenter Web installer creates duplicate alias entries in the IBM HTTP Server when you install more than one Peregrine OAA Platform application on WebSphere.

Duplicate entries can also occur if you reinstall AssetCenter Web or install another Peregrine OAA Platform application on a system that formerly had AssetCenter Web installed on it.

Remove any duplicate alias entries from the IBM HTTP Server **httpd.conf** file.

Installing WebSphere 5.0

To install WebSphere 5.0:

- 1 Make sure IIS is stopped: The **World Wide Web Publishing** service must be stopped and configured to be started automatically.
- 2 Give a user the following rights (**Local security policy/ Local Policies/ User rights assignment** menu):

- Act as part of the operating system
 - Create a token object
 - Increase quotas
 - Replace a process level token
- 3 Restart the computer.
 - 4 Start the WebSphere 5 installation program.
Select the full installation
 - 5 Set the startup mode of the **IBM WebSphere Application Server V5 - server1** service to automatic.
 - 6 Restart the computer.
 - 7 Start Websphere Administrator's Console.
 - 8 Log on to the node of your choice.
 - 9 Select **Environment\ Update Web Server Plugin**.

Configuring WebSphere 5.0

To configure WebSphere 5.0:

- 1 Set the **WAS_HOME** environment variable to the WebSphere 5.0 root folder.
By default, this folder is **C:\Program Files\WebSphere\AppServer**.
- 2 Start WebSphere 5.0.
- 3 Start the Admin console.
- 4 Create a new Enterprise Application using the **portal.<Version>.war** file located on the AssetCenter Web installation CD-ROM.
 - Populate the **Context Root** field.
Example: **/oaa** or **/getit**.
 - Keep all the other settings at their defaults.
 - Save the server configuration.
- 5 Click **Save to Master Configuration**.
- 6 Click **Save**
- 7 Run the AssetCenter Web setup program and select the **Custom** installation option.
→ [Typical installation](#) [page 58]
Clear the **Tomcat**, **JDK**, and **Apache** options.

When you are prompted for the root folder of Java, select:

C:\Program Files\WebSphere\AppServer\java.

When you are prompted for the **.war** folder, select:

**C:\Program Files\WebSphere\AppServer\installedApps\
[hostname]\portal<version>.ear\portal.<Version>.war**

- 8 When prompted for the deployment folder, indicate the WebSphere folder where the applications are installed.

Example:

```
C:\Program Files\WebSphere\AppServer\installedApps\[hostname]\portal<version>\portal.<Version>.war
```

- 9 When the installation is complete, delete **pop3.jar** from **C:\Program Files\WebSphere\AppServer\java\jre\lib\ext**.

 Note:

This file is no longer needed because WebSphere includes it in **mail.jar**.

- 10 Copy **js.jar** from **...portal.<Version>.war\WEB-INF\lib** to **C:\Program Files\WebSphere\AppServer\java\jre\lib\ext**.

- 11 In the **IBMHttpServer\conf\httpd.conf** file:

- 1 Add the following lines if they do not exist:

```
LoadModule ibm_app_server_http_module
<AppServerPath>/bin/mod_ibm_app_server_http.dll
WebSpherePluginConfig
"<AppServerPath>/config/cells/plugin-cfg.xml"
```

- 2 Add the following line if it does not already exist:

```
Alias /oaa/ "C:/Program Files/WebSphere/AppServer/installedApps/[hostname]/portal<version>/portal.<Version>.war/"
```

The installer fails to do this because earlier versions were named differently.

 Note:

This alias needs to match the **Context Root** field

→ [Deploying the portal.<Version>.war file to WebSphere \[page 41\]](#).

The path of the **portal.<Version>.war** file must be the same as the **portal.<Version>.war** file used previously in this procedure.

- 12 Add the following mapping **UriGroup** to the **plugin-cfg.xml** file in the **WebSphere\AppServer\config\cells** folder:

```

<UriGroup Name="default_host_server1_sys_Cluster_URIs">
<Uri AffinityCookie="JSESSIONID" Name="/oaa/servlet/archway"/>
<Uri AffinityCookie="JSESSIONID" Name="/oaa/servlet/rpcrouter"/>
<Uri AffinityCookie="JSESSIONID" Name="/oaa/servlet/messagerouter"/>
<Uri AffinityCookie="JSESSIONID" Name="/oaa/servlet/download/*"/>
<Uri AffinityCookie="JSESSIONID" Name="/oaa/servlet/*"/>
<Uri AffinityCookie="JSESSIONID" Name="/oaa/*.do"/>
<Uri AffinityCookie="JSESSIONID" Name="/oaa/*.jsp"/>
<Uri AffinityCookie="JSESSIONID" Name="/oaa/*.jsw"/>
<Uri AffinityCookie="JSESSIONID" Name="/oaa/*.jsw"/>
<Uri AffinityCookie="JSESSIONID" Name="/oaa/j_security_check"/>
</UriGroup>

```

- 13 Restart WebSphere 5.0.
- 14 Log in to the Admin console again.
 - Right-click **Environment** on the left part of the screen and select **Update Web Server Plugin**.
 - Click **OK**.
- 15 Restart IBM HTTP Server.
- 16 Connect to the **admin.jsp** page and configure the system.
 - ➔ [Connecting to the administration portal \[page 131\]](#).

WebSphere Translation Server

You can configure AssetCenter Web to use a WebSphere Translation Server to provide real-time translations of on-screen data.

 **Note:**

The OAA interface to the WebSphere Translation Server requires a mouse to use.

To configure WebSphere Translation Server for AssetCenter Web:

- 1 ➔ Copy the wts.jar file to the AssetCenter Web installation folder [page 51]
- 2 ➔ Configuring AssetCenter Web so that it uses WebSphere Translation Server [page 51]

Copy the wts.jar file to the AssetCenter Web installation folder

To copy the wts.jar file to the AssetCenter Web installation folder:

- 1 Stop the application server.
- 2 Go to the installation folder of WebSphere Translation Server.
- 3 Copy the **wts.jar** file.
- 4 Paste the **wts.jar** in the AssetCenter Web installation folder located in:

```
<Application server folder>\WEB-INF\lib
```

- 5 Restart the application server.

Configuring AssetCenter Web so that it uses WebSphere Translation Server

To configure AssetCenter Web so that it uses WebSphere Translation Server:

- 1 Login to the AssetCenter Web admin page (**admin.jsp**).
- 2 Click **Settings**.
- 3 Select the **Common** tab.
- 4 Populate the following parameters:
 - **Translation Server Factory Class:** Enter the Java factory class for the Translation server.
The default Java factory class is:

```
com.peregrine.util.WTSLanguageTranslatorFactory
```
 - **Language from which to translate:** Enter the source language that you want translated.
The default value is **English**.
 - **Translation Server IP Address:** Enter the IP address and communication port of Translation Server.
Example:

```
10.3.128.181:1097
```
- 5 Click **Save**.
The Control Panel is displayed.
- 6 Click **Reset Server**.

Translating on-screen data with Translation Server


If you plan to store AssetCenter Web data in a mixture of languages, you can configure AssetCenter Web to send data to a Translation Server for real time translation.

This interface will only translate data retrieved from the AssetCenter Web database or manually typed into form inputs.

If you need a translated user interface, you can purchase a language pack directly from Peregrine Systems.

To translate on-screen data with Translation Server:

- 1 Enable Translation Server from the **Administration/ Setting** page as described in section [Configuring AssetCenter Web so that it uses WebSphere Translation Server](#) [page 51].

The translation button  appears in the upper right of the toolbar.

- 2 Click the string to translate.

- 3 Click .

The translation window is displayed.

- 4 Select the target language.

The translation of your selection appears in the translation box.

WebLogic 6.1

This section explains how to make WebLogic work with AssetCenter Web.

To configure WebLogic 6.1 with IIS:

- 1 Stopping WebLogic and the Web server [page 53]
- 2 Modify the startWebLogic.cmd file [page 53]
- 3 Modifying the Server.Policy file [page 54]
- 4 Execute the AssetCenter Web installation program [page 54]
- 5 Moving .jar files to the Java development kit ext folder [page 55]
- 6 Configuring issforward.dll as an ISAPI filter and an extension [page 55]
- 7 Configuring iisproxy.dll as an extension [page 56]
- 8 Creating a virtual directory for AssetCenter Web [page 57]
- 9 Restarting WebLogic and the Web server [page 58]

Stopping WebLogic and the Web server

Before configuring WebLogic, you must stop the WebLogic server and the Web server.

To start configuring WebLogic:

- 1 Stop the WebLogic application server.
- 2 Stop the Web server.

Modify the startWebLogic.cmd file

To modify the startWebLogic.cmd file:

- 1 Open the **startWebLogic.cmd** file with a text editor.
By default, this file is located in the **c:\bea\wlserver6.1\config\<Domain name>** folder
- 2 Search the following section of the script:

```

echo *****
echo * To start WebLogic Server, use the password *
echo * assigned to the system user. The system *
echo * username and password must also be used to *
echo * access the WebLogic Server console from a web *
echo * browser. *
echo *****
@rem Set WLS_PW equal to your system password for no password prompt
server startup.
set WLS_PW=password

```

- 3 In the last line, replace **password** by the WebLogic system password.
- 4 Search the **-mx** parameter in the file.
Set a value **225 MB** and **512 MB**.

Note:

Make sure that the setting for **maximum heap size** is less than the free RAM available to the application server(s).

Exceeding the amount of available RAM causes the JVM processes to swap to disk, reducing overall performance. A setting of 256 MB should be sufficient for most systems.

- 5 Set the **STARTMODE** variable to **false**.

The first time you start WebLogic after the installation, you will need to start it in development mode for it to find the Web applications that have been deployed.

- 6 Add the following text between **javaw.exe** and **weblogic.server** (before the end of the entry **goto finish**):

```
-Djava.security.auth.login.config==<Weblogic installation folder>\lib\server.policy
```

The default installation folder of WebLogic: **c:\bea\wlserver6.1**.

- 7 Save the file.

Modifying the Server.Policy file

To modify the Server.Policy file:

- 1 Open the **Server.Policy** file in a text editor.
By default, this file is located in the **c:\bea\wlserver6.1\lib** folder.
- 2 Add the following lines to the end of the file:

```
ServerLoginModule
{
weblogic.security.internal.ServerLoginModule required debug=true;
};
```

- 3 Save the file.

Execute the AssetCenter Web installation program

Run the AssetCenter Web setup program and select the **Custom** installation option.

In the **Select Components** windows, do not select the **Tomcat** option.

The following is the list of variables you will be required to enter when prompted to by the installation program:

- **JAVA_HOME:** c:\bea\jdk131
DEPLOYMENT_DIR : bea\wlserver6.1\config**<WebLogic domain name on the computer on which WebLogic is installed>\config**<Domain name>\applications****

→ Custom installation [page 61].

Moving .jar files to the Java development kit ext folder

To move .jar files:

- 1 Create the folder `c:\bea\jdk131\jre\lib\ext` if it does not exist.
- 2 Go to the **lib** folder of the Peregrine OAA installation folder (generally `bea\wlserver6.1\config\<WebLogic domain name on the computer on which WebLogic is installed>\config\<Domain name>\applications\loaa\WEB-INF\lib`).

Move the `log4j-1.2.6.jar` file to the `\bea\jdk131\jre\lib\ext` folder.

- 3 Verify that `oaasecurityproxy.jar` is in the `\bea\jdk131\jre\lib\ext` folder.
- 4 Go to the Peregrine OAA **external** folder (typically `Peregrine\loaa\external`) and copy the following files to the `\bea\jdk131\jre\lib\ext` folder:

- `jaas.jar`
- `jai_codec.jar`
- `jai_core.jar`
- `jce1_2_1.jar`
- `jcrt.jar`
- `jnet.jar`
- `jsse.jar`
- `local_policy.jar`
- `mllibwrapper_jai.jar`
- `sunjce_provider.jar`
- `US_export_policy.jar`

Configuring issforward.dll as an ISAPI filter and an extension

To establish a connection between WebLogic and IIS, you will need to install the `iisforward.dll` file as an ISAPI filter.

To install issforward.dll as an ISAPI filter and an extension:

- 1 Start the Windows Internet Services management console (**Start/ Settings/ Control Panel/ Administrative Tools/ Internet Services (IIS)** in Windows XP).
- 2 Right click the **Default Web Site** node, then select **Properties**.
A dialog box is displayed.

- 3 Display the **ISAPI Filters** tab.
- 4 Click **Add**.
- 5 Enter the following information:
 - **Filter name:** `iisforward`.
 - **Executable:** `issforward.dll`.
The default path is `c:\bea\wlservlet6.1\bin\issforward.dll`.
- 6 Click **OK**.
- 7 Display the **Home Directory** tab.
- 8 Click **Configuration**.
The **Application Configuration** page opens on the **App Mappings** tab.
- 9 Verify that there is a mapping for the `.wforward` extension.
If it does not exist, click **Add** to map the `.wforward` extension.
- 10 Enter the following information:
 - **Executable:** `issforward.dll`.
By default, `issforward.dll` is located in the folder `c:\bea\wlservlet6.1\bin`.
 - **Extension:** `.wforward`.
- 11 Exit the Internet Services management console.

Configuring `iisproxy.dll` as an extension

To establish a connection between WebLogic and IIS, you will need to install the `iisproxy.dll` file as an extension.

To configure `iisproxy.dll` as an extension:

- 1 Start the Internet Services management console.
- 2 Right click the **Default Web Site** node, then select **Properties**.
- 3 Display the **Home Directory** tab.
 - Select the option **Read**.
 - From the **Execute Permissions** drop-down list, select **Scripts and Executables**
- 4 Click **Configuration**.
The **Application Configuration** page opens on the **App Mappings** tab.
- 5 Verify that there is a mapping for the `.jsp` extension and that it is mapped to `c:\bea\wlservlet6.1\bin\iisproxy.dll`.
If this mapping does not exist:

- 1 Click **Add**.
- 2 Enter the following information:
 - **Executable: iisproxy.dll.**
By default, **iisproxy.dll** is located in the folder **c:\bea\wlsrver6.1\bin**.
 - **Extension: .jsp.**



Note:

Make sure the **iisproxy.dll** and **iisforward.dll** files are located in the same folder.

Creating a virtual directory for AssetCenter Web

In order to run AssetCenter Web, you will need to create a virtual directory on the Web server.

The virtual directory must point to the WebLogic installation folder.

To create the virtual folder:

- 1 Start the Windows Internet Services management console (**Start/ Settings/ Control Panel/ Administrative Tools/ Internet Services (IIS)** menu in Windows XP).
- 2 Right-click the **Default Web Site** node and select the **New/ Virtual directory** menu.
- 3 Create the virtual directory using the following parameters:

Table 3.4. WebLogic in Windows - Creating a virtual directory

Setting	Value
Virtual directory alias	<Name of the virtual directory to use for AssetCenter Web> This name must be in the configuration of the application server. We recommend using the name oaa .
Web site content directory	<WebLogic installation folder>\config<Domain name>\applications\oaa By default, WebLogic is installed to c:\bea\wlsrver6.1\config<Domain name>\applications\oaa .
Access authorizations	Read, Execute scripts

- 4 Right-click the virtual directory you have just created under **Default Web Site**.
Select **Properties** and verify that the permission is **Scripts and Executables**.
- 5 Click **Configuration** on the **Home Directory** tab.
The **Application Configuration** page opens on the **App Mappings** tab.
- 6 Verify that there is a mapping for the **.jsp** extension and that it is mapped to **c:\bea\wlserver6.1\bin\iisproxy.dll**.

Restarting WebLogic and the Web server

You must restart the servers in order to enable the new WebLogic configurations.

To take into account the new WebLogic configuration:

- 1 Restart the Web server.
- 2 Restart the WebLogic server.
- 3 Start AssetCenter Web.

Typical installation

A **Typical** installation of AssetCenter Web installs the most commonly used components of the product and saves application files and data in default destination directories.

Most AssetCenter Web users choose **Typical** installation.

Components

The following table lists the components automatically installed with a **Typical** installation:

Components and installation folder

Table 3.5. Typical installation in Windows - components

AssetCenter Web component	Default installation folder
Apache Web Server	C:\Program Files\Peregrine\Common\Apache2
Tomcat Application Server	C:\Program Files\Peregrine\Common\Tomcat4
Java Development Kit	C:\Program Files\Peregrine\Common\jdk1.3.1_05
OAA platform with the AssetCenter Web pages	C:\Program Files\Peregrine\oaa

Services

The installation program also creates and starts the following services on your server:

- Apache
- Peregrine Tomcat

Important:

If you are already running another Web or application server on the target system, you will need to stop the services for the applications before beginning the AssetCenter Web setup

If you do not stop these services before beginning the installation process, the setup program will complete successfully, however, it may not create the Apache and Peregrine Tomcat services needed to run AssetCenter Web.

Communication ports

The **Typical** installation uses the following ports:

Table 3.6. Typical installation in Windows - ports

Default port	Component that uses it	Alternate port
80	Apache Web Server	8081
8005	Tomcat application server administration	8015
8009	Tomcat application server worker	8019

Default port	Component that uses it	Alternate port
8011	Tomcat application server worker file for load balancing (optional)	8021
8013	Tomcat application server worker file for load balancing (optional)	8023
8015	Tomcat application server worker file for load balancing (optional)	8025

After installation, you must configure AssetCenter Web to use alternate communication ports, in case the standard ports are already used.



Note:

To change the settings of these packages, or to use or install other packages, execute the AssetCenter Web installation program with the **Typical** option.

Typical installation procedure

This section explains how to perform a standard installation of AssetCenter Web.

This installation mode presupposes that you use Tomcat as the application server and Apache as the Web server.

If you decide to use different servers: → [Custom installation](#) [page 61].



Warning:

If you were using a previous version of AssetCenter Web, you must uninstall this version before installing AssetCenter Web 4.3. When uninstalling, OAA, Apache, Tomcat and SDK Java2 Standard Edition are automatically uninstalled.

To install AssetCenter Web, Apache, Tomcat and JDK:

- 1 Insert the AssetCenter CD-ROM.
- 2 If the installation program's graphical window does not automatically appear when you insert the CD-ROM:
 - 1 Execute the Windows explorer.
 - 2 Select the CD-ROM.
 - 3 Select the root of the CD-ROM.
 - 4 Execute the **autorun.exe** program.

- 3 Select **AssetCenter Web** and then click **Install**.
- 4 Select one of the **Typical** modes:
The installation program copies the necessary files, then creates and starts the **Apache** and **Peregrine Tomcat** services.
- 5 If you install AssetCenter Web on a computer on which Tomcat is already installed, modify the **JAVA_HOME** manually so that points to the new JDK (Java Development Kit).

In effect, the installation program does not update the **JAVA_HOME** environment.

By default, the JDK folder is:

```
C:\Program Files\Peregrine\Common\jdk1.3.1_05
```

- 6 Start the Windows Services console (**Start/ Programs/ Administrative Tools/ Services** menu in Windows XP).
Make sure the **Apache** and **Peregrine Tomcat** services are started. (The status of these services must be **Started**).
In case of problem: → [Troubleshooting](#) [page 155].

After executing this procedure, **AssetCenter Web** is fully installed in the standard fashion.

Custom installation

This section describes the custom installation of AssetCenter Web.

Components

Compatible applications

For the list of supported packages: → [Supported environments](#) [page 19].

Communication ports

The communication ports used by the AssetCenter Web installation program depend on the packages that you select. Refer to the documentation of your Web server and your application server to determine which communication ports they require.

Table 3.7. Custom installation in Windows - ports

Default port	Component that uses it	Alternate port
8005	Tomcat application server administration	8015
8009	Tomcat application server worker	8019
80	IIS Web Server	8081

After installation, you must configure AssetCenter Web to use alternate communication ports, in case the standard ports are already used.

Port conflicts

If you are using Tomcat as application server with Oracle 9.2.0.1, then, by default, there will be a conflict with port 8009. To resolve this conflict, you have the following solutions available:

- Install Oracle 9.2.0.1 on a separate server (recommended).
- Configure Oracle so that it uses a different port.
- Configure the Tomcat instances so that they use a different port.

For more information on how to configure the Tomcat ports: → [Load balancing for application servers in Windows](#) [page 103].

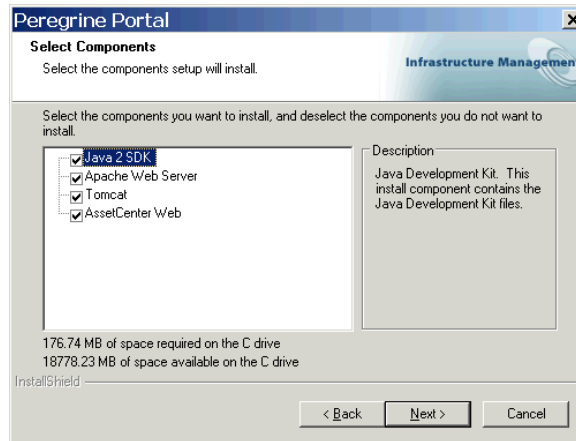
For further information on resolving port conflicts: → [Checking for Tomcat port conflicts](#) [page 162].

Custom installation procedure

To perform a custom installation of AssetCenter Web:

- 1 Insert the AssetCenter CD-ROM.
- 2 If the installation program's graphical window does not automatically appear when you insert the CD-ROM:
 - 1 Execute the Windows explorer.
 - 2 Select the CD-ROM.
 - 3 Select the root of the CD-ROM.
 - 4 Execute the **autorun.exe** program.
- 3 Select AssetCenter Web and then click **Install**.

- 4 Select the **Custom** installation mode.
- 5 Here is some information about the screens that might raise questions:
 - **Figure 3.5. Custom installation in Windows - selecting packages**



Select the packages to be installed and **deselect those you do not want to install.**

For a development environment: Deselect the packages that you will install manually or that you are going to replace by another application. For example, deselect **Apache Web Server** if you are going to replace it by another Web server.

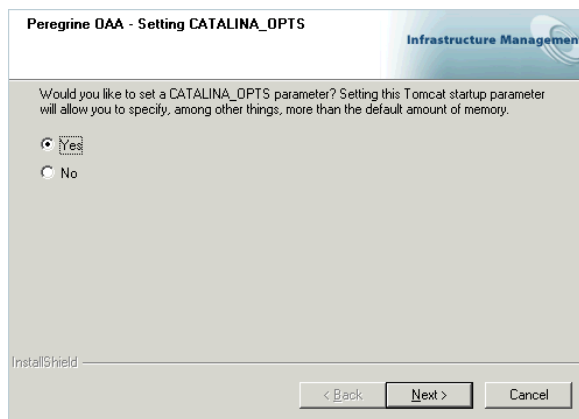
For a production environment: select the components to install on the current computer:

Table 3.8. Custom installation in Windows - Packages

Component	Comment
Java 2 SDK	Installs Sun Microsystems Java Development Kit (Java 2 SDK). You must install this package on each computer you are going to install an application server.
Apache Web Server	AssetCenter Web requires a Web server in order to serve up pages.

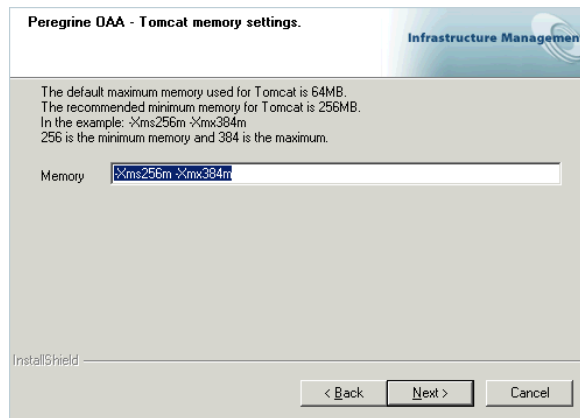
Component	Comment
Tomcat	<p>Installs the Tomcat application server.</p> <p>AssetCenter Web requires at least one application server in order process Java applications. You can install multiple instances of the application server for load balancing.</p> <p>You must install this package on the same computer as the AssetCenter Web package.</p>
AssetCenter Web	<p>Copies the files needed by AssetCenter Web. You must install these files on the same computer as the primary application server.</p>

- **Figure 3.6. Custom installation in Windows - CATALINA_OPTS**



Select **Yes** if you wish to configure the Tomcat memory parameters.

- **Figure 3.7. Custom installation in Windows - Memory settings for Tomcat**



Populate the **Memory** field with, for example, the value **-Xms256m -Xmx384m**.

- 6 The installation program copies the required files, then creates and starts the corresponding services on the application server and on the Web server.
- 7 Start the Windows Services console (**Start/ Programs/ Administrative Tools/ Services** menu in Windows XP).

Make sure the **Apache** and **Peregrine Tomcat** services are started. (The status of these services must be **Started**).

In case of problem: → [Troubleshooting](#) [page 155].

After executing this procedure, AssetCenter Web is fully installed and customized.

Uninstallation

This section describes the uninstallation procedure for AssetCenter Web.

 **Warning:**

This procedure uninstalls all the components that you installed with the installation program.

If you selected the **Typical** installation, all components will automatically be uninstalled.

If you selected the **Custom** installation, only the components that you selected will be uninstalled.

To uninstall AssetCenter Web:

- 1 Start the Windows Add or Remove Programs console (**Start/ Settings/ Control Panel/ Add or Remove Programs**, in Windows XP).
 - 2 Select the program **Peregrine Portal 4.1.** and then click **Modify/ Delete.**
 - 3 Follow the indications given by the uninstallation program.
-

 **Note:**

If there are any shared files to be removed, a message window appear.

If WebSphere is installed on this computer, the uninstallation program prompts you to confirm deletion of the **.jar** files.

You must not accept to delete these files because they are required by WebSphere Advanced Administrative Console.

Tests

To verify that AssetCenter Web is correctly installed:

- 1 Check that the Web server and the application server are started.
- 2 Start a Web browser.
- 3 Enter the following URL:

```
http://<Name of the Web server>:<Port>/oaa/admin.jsp
```

The standard port number depends on the application server used:

Table 3.9. Installing in Windows - Standard port numbers

Application server used	Standard port number
WebSphere	9080
WebLogic	7001
Tomcat	80, may be omitted from the URL

If everything is correctly installed and configured, the opening administration page is displayed.

If the opening page of the administration session does not display: → [Troubleshooting](#) [page 155].

4 Installing in UNIX

CHAPTER

This chapter explains how to install and configure AssetCenter Web in UNIX.

Using standard application (Tomcat) and Web (Apache) servers

If you wish to use Tomcat and Apache, and they are not yet installed on the AssetCenter Web server(s), you can read section [Typical installation](#) [page 86] directly.

Using an alternative application or Web server

AssetCenter Web requires a Web server and a Java-enabled application server to run.

For the list of supported servers: → [Supported environments](#) [page 19].

By default, the installation program installs Tomcat and Apache.

If you already use other combinations of servers, you must install and configure these **before** executing the AssetCenter Web installation program.

If you already have the Apache and Tomcat servers

If you use the **Typical** installation option, the AssetCenter Web installer configures Tomcat to connect to a new instance of the Apache Web server. If Apache and Tomcat are already installed, you can configure AssetCenter Web so that it uses these external applications by copying the required files using the **Typical** installation.

To configure an existing Tomcat server so that it connects to an Apache server:

- 1 Execute the AssetCenter Web installation program.

Select the **Typical** installation type.

→ [Typical installation](#) [page 58].

The Tomcat and Apache instances installed provide the necessary source files to copy.



Important:

Do not start these instances of Tomcat and Apache that have just been installed.

- 2 Stop existing instances of both Tomcat and Apache.
- 3 Copy the following files:

Table 4.1. Using an existing Apache or Tomcat server in Unix - Files to copy

File created in Typical installation of Apache and Tomcat	Destination
mod_jk.conf	/Conf sub-folder of existing Tomcat installation folder. Example: /usr/local/peregrine/common/Tomcat 4/conf .
workers.properties	/Conf sub-folder of existing Tomcat installation folder. Example: /usr/local/peregrine/common/Tomcat 4/conf .
mod_jk.dll	/modules sub-folder of existing Apache installation folder. Example: /usr/local/peregrine/common/apache2/modules .



Note:

The **mod_jk.conf** file provided with AssetCenter Web is compatible with Apache 2.0.43 and Tomcat 4.1.25.

- 4 Using a text editor, open the files **mod_jk.conf** and **workers.properties**. (These files are located in the **conf** folder of the existing Tomcat server.)
Make sure the access paths of Tomcat point to the existing version of (before the **Typical** installation).

Do the same for the JDK access paths.

- 5 Using a text editor, open the **httpd.conf** file. (This file is located in the **/conf** folder of the existing Apache installation).
 - Add the access path of the existing version of Tomcat (before the **Typical** installation) to the **Global Environment** section, **include** clause:

```
### Section 1: Global Environment
include "<Existing absolute path of Tomcat>/conf/mod_jk.conf"
```

- Add **login.jsp** to the list of files in the **DirectoryIndex** section:

```
# DirectoryIndex: Name of the file or files to use as a pre-written
# HTML directory index. Separate multiple entries with spaces.
#
<IfModule mod_dir.c>
DirectoryIndex index.html login.jsp
</IfModule>
```

- Add the following line to the end of the file:

```
Alias <Access path of Tomcat>/webapps/oa
```

- 6 Execute the AssetCenter Web installation program.
Select the **Custom** installation type.
→ [Custom installation \[page 91\]](#).
- 7 Start Apache and Tomcat.
- 8 Start a Web browser and enter the address of the AssetCenter Web login page.
→ [Connecting to AssetCenter Web \[page 144\]](#).
Make sure you can connect.

If the address of the AssetCenter Web login page displays a list of files instead of the login page:

- 1 In a text editor, open the **<Apache installation folder>/conf/httpd.conf**.
- 2 Search for the string `DirectoryIndex index.html`.
- 3 Add `login.jsp` to the end of the line, which gives:

```
DirectoryIndex index.html login.jsp
```

- 4 Save the modified file.
- 5 Restart Apache.

WebSphere 4.0

This section describes the procedure to configure WebSphere 4.0 to be used as a substitute for Apache and Tomcat.



Note:

The AssetCenter Web installer creates duplicate alias entries in the IBM HTTP Server when you install more than one Peregrine OAA Platform application on WebSphere.

Duplicate entries can also occur if you reinstall AssetCenter Web or install another Peregrine OAA Platform application on a system that formerly had AssetCenter Web installed on it.

Remove any duplicate alias entries from the IBM HTTP Server **httpd.conf** file.

To configure WebSphere 4.0:

- 1 Installing WebSphere 4.0 [page 73]
- 2 Deploying the portal.<Version>.war file to WebSphere [page 73]
- 3 Setting the heap size parameter [page 76]
- 4 Configuring a virtual directory for IBM HTTP Server [page 77]
- 5 Execute the AssetCenter Web installation program [page 78]

If you plan on setting up WebSphere Translation Server: → WebSphere Translation Server [page 81].

Installing WebSphere 4.0



Tip:

To check which version of WebSphere and the fix pack are installed, go to the **opt/WebSphere/AppServer/logs/default_server_Stdout.log** file.



Note:

WebSphere includes IBM HTTP Server.

Deploying the portal.<Version>.war file to WebSphere

The **portal.<Version>.war** file creates the folder structure necessary to deploy AssetCenter Web with the application server.

Once this file is deployed, it is possible to execute the AssetCenter Web installation program.

To deploy the portal.<Version>.war file to WebSphere:

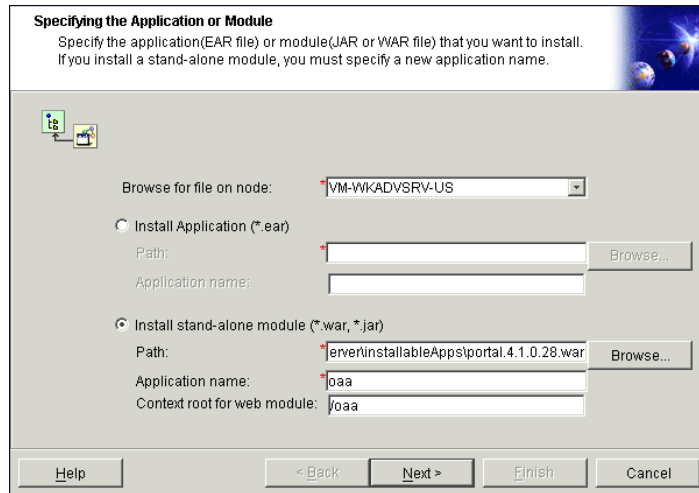
- 1 Verify that the WebSphere Admin Server has been started.
- 2 Start the WebSphere Advanced Administrator's Console (**Start/ Programs/ IBM WebSphere/ Application Server/ Administrator's Console** menu in Windows XP).
- 3 On the menu at the left side of the console, right-click **Enterprise Applications** and select **Install Enterprise Application**.
- 4 In the window that is displayed:
 - Select **Install stand-alone module**.
 - **Path** field: Browse to the folder containing the **portal<most recent version>.war** file.
 - **Application Name** field: Type **oaa**.
 - **Context Root** field: Type the name of the virtual Web server folder you wish to associate with AssetCenter Web.

Example: /oaa.



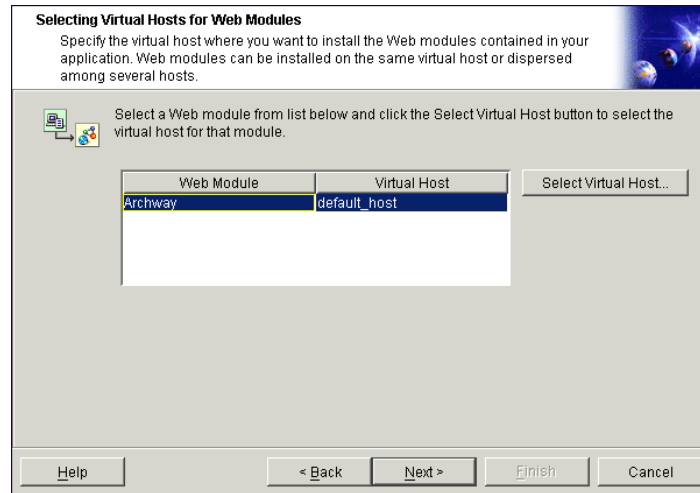
Important:

You must create a virtual directory matching the path you enter here.

Figure 4.1. WebSphere in Windows - Deploying the portal.<Version>.war file

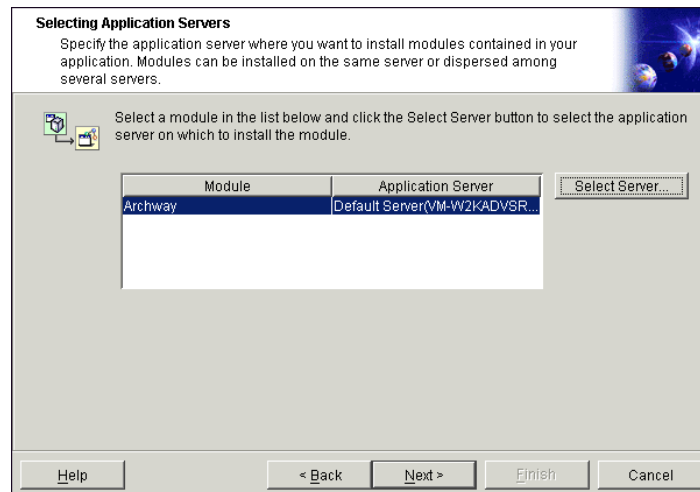
- 5 Click **Next** without modifying the following screens:
 - Mapping EJB Run As Roles to Users
 - Binding Enterprise Beans to JNDI Names
 - Mapping EJB References to Enterprise Beans
 - Mapping Resource References to Resources
 - Specifying the Default Datasource
 - Specifying Data Sources for Individual CMP Beans
- 6 In the **Selecting Virtual Hosts for Web Modules** window, select the instance of the WebSphere server to use, and then click **Next**.

Figure 4.2. WebSphere in Unix - Selecting the virtual folder



- 7 In the **Selecting Application Servers** folder, select the instance of the WebSphere server to use, and then click **Next**.

Figure 4.3. WebSphere in UNIX - Selecting the application server



- 8 Click **Finish**.
- 9 Go to **nodes/<Your computer>/Application servers**.
In the **JVM** tab, specify **60** and **225**.
- 10 Close all windows.
- 11 Restart the computer.

Setting the heap size parameter

You can configure how much memory is made available for your application server.

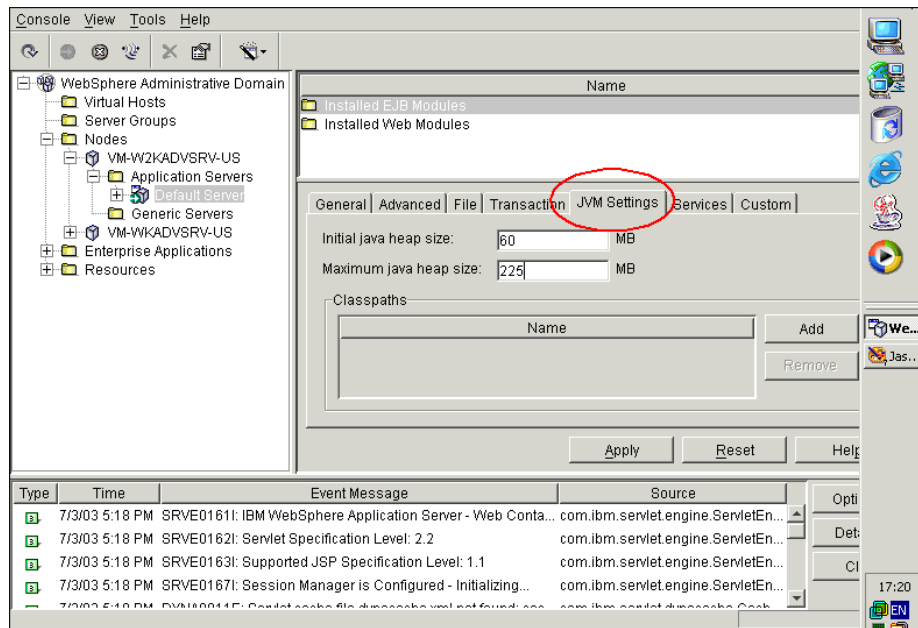
The following instructions assume you are only using one WebSphere instance.

You will need to adjust the **heap size** parameter accordingly if you are load balancing across several WebSphere instances.

To set the heap size parameter:

- 1 Verify that the WebSphere Admin Server has been started.
- 2 Open the WebSphere Advanced Administrator's Console (**Start/ Programs/ IBM WebSphere/ Application Server/ Administrator's Console** in Windows XP).
- 3 Select **Nodes/ <System Name>/ Application Servers/ <Application server name>**.
- 4 Click the **JMV settings** tab:

Figure 4.4. WebSphere in UNIX - JVM settings tab



Populate the following fields:

- **Initial java heap size** field: Type **60**.
- **Maximum java heap size** field: Type the value you want for heap memory.

This setting should be at least 225 MB, but not more than 512 MB.

 **Note:**

Make sure that the setting for **maximum heap size** is less than the free RAM available to the application server(s).

Exceeding the amount of available RAM causes the JVM processes to swap to disk, reducing overall performance. A setting of 256 MB should be sufficient for most systems.

Configuring a virtual directory for IBM HTTP Server

You must configure a virtual directory for AssetCenter Web in your Web server.

The following instructions assume that you are using IBM HTTP Server as the Web server (WebSphere's default Web server).

See your Web server documentation to determine how to create a virtual directory if you are using another Web server.

To configure a virtual directory for IBM HTTP Server:

- 1 Stop IBM HTTP Server.
- 2 Open the **httpd.conf** file in a text editor.
By default, this file is located in the **<root>/usr/HTTPServer/conf** folder.
- 3 Add the following line to the end of the file:

```
Alias /oaa/ "<root>/WebSphere/AppServer/installedApps/oaa.ear/porta  
l.<Version of .war file installed>.war/"
```

Important:

The name of the virtual folder must match the value of the **Context Root** field.

→ [Deploying the portal.<Version>.war file to WebSphere \[page 73\]](#).

- 4 Save the file.
- 5 Start IBM HTTP Server.

Execute the AssetCenter Web installation program

Execute the installation program and select the **Custom** installation option.

→ [Custom installation \[page 91\]](#).

Note:

If needed, you can regenerate the plugin configuration using the Websphere Admin console after running the AssetCenter Web installer.

If you plan on setting up WebSphere Translation Server: → [WebSphere Translation Server \[page 81\]](#).

Regenerating the plug-in configuration

To regenerate the plug-in configuration

- 1 Start the WebSphere Advanced Administrator's Console (**Start/ Programs/ IBM WebSphere/ Application Server/ Administrator's Console** menu in Windows XP).
- 2 Select **Nodes/ <System Name>/ Application Servers/ <Application server name>**.
- 3 Right-click **<System Name>**, and then select **Regen Webserver Plugin**.
Copy the following lines from the **<settings>** section of the **<Location of the application server>/WEB-INF/default/archway.xml** to the **<settings>** section of the **<Location of the application server>/WEB-INF/local.xml** file:


```
<SSLProvider>com.ibm.jsse.JSSEProvider</SSLProvider>
<HTTPSHandlerPkg>com.ibm.net.ssl.internal.www.protocol</HTTPSHandlerPkg>
<CryptoProvider>com.ibm.crypto.provider.IBMJCE</CryptoProvider>
```
- 4 Restart your application server.

WebSphere 5.0

This section describes the procedure to follow to configure WebSphere 5.0 to replace Apache and Tomcat.



Note:

The AssetCenter Web installer creates duplicate alias entries in the IBM HTTP Server when you install more than one Peregrine OAA Platform application on WebSphere.

Duplicate entries can also occur if you reinstall AssetCenter Web or install another Peregrine OAA Platform application on a system that formerly had AssetCenter Web installed on it.

Remove any duplicate alias entries from the IBM HTTP Server **httpd.conf** file.

To configure WebSphere 5.0:

- 1 Set the **WAS_HOME** environment variable to the WebSphere 5.0 root folder.
By default, this folder is **/opt/WebSphere/AppServer**.
- 2 Start WebSphere 5.0.
- 3 Start the Admin console.

- 4 Create a new Enterprise Application using the **portal.<Version>.war** file located on the AssetCenter Web installation CD-ROM.
 - Populate the **Context Root** field.
Example: **/oaa** or **/getit**.
 - Keep all the other settings at their defaults.
 - Save the server configuration.
- 5 Run the AssetCenter Web setup program and select the **Custom** installation option.

→ [Custom installation](#) [page 91].

Clear the **Tomcat**, **JDK**, and **Apache** options.

When you are prompted for the root folder of Java, select:

/opt/WebSphere/AppServer/java.

When you are prompted for the **.war** folder, select:

**/opt/WebSphere/AppServer/installedApps/
[hostname]/portal<version>.ear/portal.<Version>.war**

- 6 When prompted for the deployment folder, indicate the WebSphere folder where the applications are installed.

Example:

```
/opt/WebSphere/AppServer/installedApps/[hostname]/portal<version>/portal.<Version>.war
```

- 7 When the installation is complete, delete **pop3.jar** from **/opt/WebSphere/AppServer/java/jre/lib/ext.**

 **Note:**

This file is no longer needed because WebSphere includes it in **mail.jar**.

- 8 Copy **js.jar** from **...portal.<Version>.war\WEB-INF\lib** to **/opt/WebSphere/AppServer/java/jre/lib/ext.**
- 9 In the **IBMHttpServer/conf/httpd.conf** file:

- 1 Add the following lines if they do not exist:

```
LoadModule ibm_app_server_http_module
<AppServerPath>/bin/mod_ibm_app_server_http.dll
WebSpherePluginConfig
"<AppServerPath>/config/cells/plugin-cfg.xml"
```

- 2 Add the following line if it does not already exist:


```
Alias /oaa/ "/opt/WebSphere/AppServer/installedApps/[hostname]/portal<version>/portal.<Version>.war/"
```

The installer fails to do this because earlier versions were named differently.

 **Note:**

This alias needs to match the **Context Root** field

→ Deploying the portal.<Version>.war file to WebSphere [page 73].

The path of the **portal.<Version>.war** file must be the same as the **portal.<Version>.war** file used previously in this procedure.

- 10 Add the following mapping **UriGroup** to the **plugin-cfg.xml** file in the **/opt/WebSphere/AppServer/config/cells** folder:

```
<UriGroup Name="default_host_server1_sys_Cluster_URIs">
<Uri AffinityCookie="JSESSIONID" Name="/oaa/servlet/archway"/>
<Uri AffinityCookie="JSESSIONID" Name="/oaa/servlet/rprouter"/>
<Uri AffinityCookie="JSESSIONID" Name="/oaa/servlet/messagerouter"/>
<Uri AffinityCookie="JSESSIONID" Name="/oaa/servlet/download/*"/>
<Uri AffinityCookie="JSESSIONID" Name="/oaa/servlet/*"/>
<Uri AffinityCookie="JSESSIONID" Name="/oaa/*.do"/>
<Uri AffinityCookie="JSESSIONID" Name="/oaa/*.jsp"/>
<Uri AffinityCookie="JSESSIONID" Name="/oaa/*.jsw"/>
<Uri AffinityCookie="JSESSIONID" Name="/oaa/*.jsw"/>
<Uri AffinityCookie="JSESSIONID" Name="/oaa/j_security_check"/>
</UriGroup>
```

- 11 Restart WebSphere 5.0.
- 12 Log in to the Admin console again.
 - Right-click **Environment** on the left part of the screen and select **Update Web Server Plugin**.
 - Click **OK**.
- 13 Restart IBM HTTP Server.
- 14 Connect to the **admin.jsp** page and configure the system.
 - Connecting to the administration portal [page 131].

WebSphere Translation Server

You can configure AssetCenter Web to use a WebSphere Translation Server to provide real-time translations of on-screen data.



Note:

The OAA interface to the WebSphere Translation Server requires a mouse to use.

To configure WebSphere Translation Server for AssetCenter Web:

- 1 → Copy the `wts.jar` file to the AssetCenter Web installation folder [page 82]
- 2 → Configuring AssetCenter Web so that it uses WebSphere Translation Server [page 82]

Copy the `wts.jar` file to the AssetCenter Web installation folder

To copy the `wts.jar` file to the AssetCenter Web installation folder:

- 1 Stop the application server.
- 2 Go to the installation folder of WebSphere Translation Server.
- 3 Copy the **`wts.jar`** file.
- 4 Paste the **`wts.jar`** in the AssetCenter Web installation folder located in:

```
<Application server folder>/WEB-INF/lib
```

- 5 Restart the application server.

Configuring AssetCenter Web so that it uses WebSphere Translation Server

To configure AssetCenter Web so that it uses WebSphere Translation Server:

- 1 Login to the AssetCenter Web administration page (**`admin.jsp`**).
- 2 Click **Settings**.
- 3 Select the **Common** tab.
- 4 Populate the following parameters:
 - **Translation Server Factory Class:** Enter the Java factory class for the Translation server.

The default Java factory class is:

```
com.peregrine.util.WTSLanguageTranslatorFactory
```

- **Language from which to translate:** Enter the source language that you want translated.

The default value is **English**.

- **Translation Server IP Address:** Enter the IP address and communication port of Translation Server.

Example:

10.3.128.181:1097

- 5 Click **Save**.

The Control Panel is displayed.

- 6 Click **Reset Server**.

Translating screen data with Translation Server


If you plan to store AssetCenter Web data in a mixture of languages, you can configure AssetCenter Web to send data to a Translation Server for real time translation.

This interface will only translate data retrieved from the AssetCenter Web database or manually typed into form inputs.

If you need a translated user interface, you can purchase a language pack directly from Peregrine Systems.

To translate on-screen data with Translation Server:

- 1 Enable Translation Server from the **Administration/ Settings** page following the instructions given in the section → [Configuring AssetCenter Web so that it uses WebSphere Translation Server](#) [page 82].

The translation button  appears in the upper right of the toolbar.

- 2 Click the string to translate.

- 3 Click .

The translation window is displayed.

- 4 Select the target language.

The translation of your selection appears in the translation box.

WebLogic 6.1

This section explains how to make WebLogic work with AssetCenter Web.

To configure WebLogic 6.1 with IIS:

- 1 → Stopping WebLogic and the Web server [page 84]
- 2 → Modify the startWebLogic.cmd file [page 84]
- 3 → Execute the AssetCenter Web installation program [page 85]
- 4 → Creating a virtual directory for AssetCenter Web [page 85]
- 5 → Restarting WebLogic and the Web server [page 86]

Stopping WebLogic and the Web server

Before configuring WebLogic, you must stop the WebLogic server and the Web server.

To start configuring WebLogic:

- 1 Stop the WebLogic application server.
- 2 Stop the Web server.

Modify the startWebLogic.cmd file

To modify the startWebLogic.cmd file:

- 1 Open the **startWebLogic.cmd** file with a text editor.
By default, this file is located in the folder **/bea/wlserver6.1/config/<Domain name>**
- 2 Search the following section of the script:

```

echo *****
echo * To start WebLogic Server, use the password *
echo * assigned to the system user. The system *
echo * username and password must also be used to *
echo * access the WebLogic Server console from a web *
echo * browser. *
echo *****
@rem Set WLS_PW equal to your system password for no password promp
t
server startup.
set WLS_PW=password

```

- 3 In the last line, replace **password** by the WebLogic system password.
- 4 Search the **-mx** parameter in the file.
Set a value **225 MB** and **512 MB**.



Note:

Make sure that the setting for **maximum heap size** is less than the free RAM available to the application server(s).

Exceeding the amount of available RAM causes the JVM processes to swap to disk, reducing overall performance. A setting of 256 MB should be sufficient for most systems.

-
- 5 Set the **STARTMODE** variable to **false**.

The first time you start WebLogic after the installation, you will need to start it in development mode for it to find the Web applications that have been deployed.

- 6 Save the file.

Execute the AssetCenter Web installation program

Run the AssetCenter Web setup program and select the **Custom** installation option.

→ [Custom installation](#) [page 91].

Creating a virtual directory for AssetCenter Web

In order to run AssetCenter Web, you will need to create a virtual directory on the Web server.

The virtual directory must point to the WebLogic installation folder.

To create the virtual folder:

- 1 Start the Windows Internet Services management console (**Start/ Settings/ Control Panel/ Administrative Tools/ Internet Services (IIS)** menu in Windows XP).
- 2 Right-click the **Default Web Site** node and select the **New/ Virtual directory** menu.
- 3 Create the virtual directory using the following parameters:

Table 4.2. WebLogic in Unix - Creating a virtual directory

Setting	Value
Virtual directory alias	<Name of the virtual directory to use for AssetCenter Web> This name must be in the configuration of the application server. We recommend using the name oaa .
Web site content directory	<WebLogic installation folder>/config/<Domain name>/applications/oaa By default, WebLogic is installed in the /bea/wlserver6.1/config/<Domain name> folder.
Access authorizations	Read, Execute scripts

Restarting WebLogic and the Web server

You must restart the servers in order to enable the new WebLogic configurations.

To take into account the new WebLogic configuration:

- 1 Restart the Web server.
- 2 Restart the WebLogic server.
- 3 Start AssetCenter Web.

Typical installation

A **Typical** installation of AssetCenter Web installs the most commonly used components of the product and saves application files and data in default destination directories.

Note:

If you interrupt the installation before the end of the procedure, you must run the uninstallation program to delete the files that have been created.

Most AssetCenter Web users choose **Typical** installation.

Packages of the Typical installation type

The following table lists the components automatically installed with a **Typical** installation:

Components and installation folder

Table 4.3. Typical installation in UNIX - packages

AssetCenter Web component	Default installation folder
Apache Web Server	/usr/local/peregrine/common/apache2
Tomcat Application Server	/usr/local/peregrine/common/tomcat4
Java Development Kit	/usr/local/peregrine/common/jdk1.3.1
OAA platform with the AssetCenter Web pages	/usr/local/peregrine/oa

Communication ports

The **Typical** installation uses the following ports:

Table 4.4. Typical installation in UNIX - ports

Default port	Component that uses it	Alternate port
80	Apache Web Server	8081
8005	Tomcat application server administration	8015
8009	Tomcat application server worker	8019
8011	Tomcat application server worker file for load balancing (optional)	8021
8013	Tomcat application server worker file for load balancing (optional)	8023
8015	Tomcat application server worker file for load balancing (optional)	8025

After installation, you must configure AssetCenter Web to use alternate communication ports, in case the standard ports are already used.



Note:

To change the settings of these packages, or to use or install other packages, execute the AssetCenter Web installation program with the **Custom** option.

→ [Custom installation](#) [page 91].

Typical installation procedure

This section explains how to perform a standard installation of AssetCenter Web.

This installation mode presupposes that you use Tomcat as the application server and Apache as the Web server.

If you decide to use different servers: → [Custom installation](#) [page 91].



Warning:

If you were using a previous version of AssetCenter Web, you must uninstall this version before installing AssetCenter Web 4.3. When uninstalling, OAA, Apache, Tomcat and SDK Java2 Standard Edition are automatically uninstalled.

To install AssetCenter Web, Apache, Tomcat and JDK:

- 1 Log on to the server using an account that has **root** rights.
-



Important:

Make sure your temporary folder has at least 300 MB of free disk space.

In Solaris, for example, the global temporary folder is **/tmp**.

- 2 Insert the AssetCenter CD-ROM.
- 3 The installation program should start automatically.
Exit the installation and then mount the CD-ROM directly.

Example: `mount /cdrom`

Go to the CD-ROM drive.

Example: `cd /cdrom`.

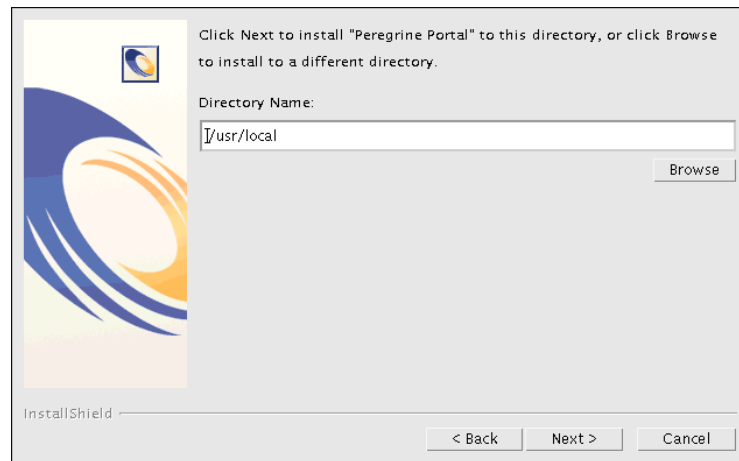
Enter the installation script corresponding to your operating system:

Table 4.5. Typical installation in UNIX - script to execute

Operating system	Command interpreter script to execute
AIX	./setupaix
Red Hat Linux	./setuplinux
Solaris	./setupsolaris

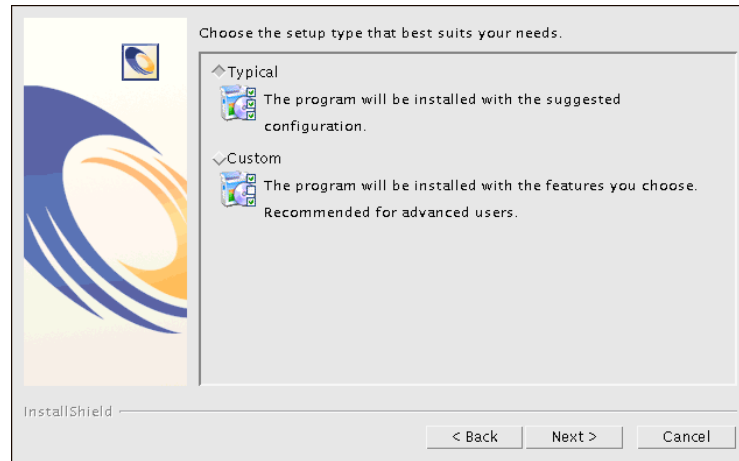
The welcome page of the installation program is displayed.

- 4 Here is some information about the screens that might raise questions:
 - **Figure 4.5. Typical installation in UNIX - Peregrine Portal folder**



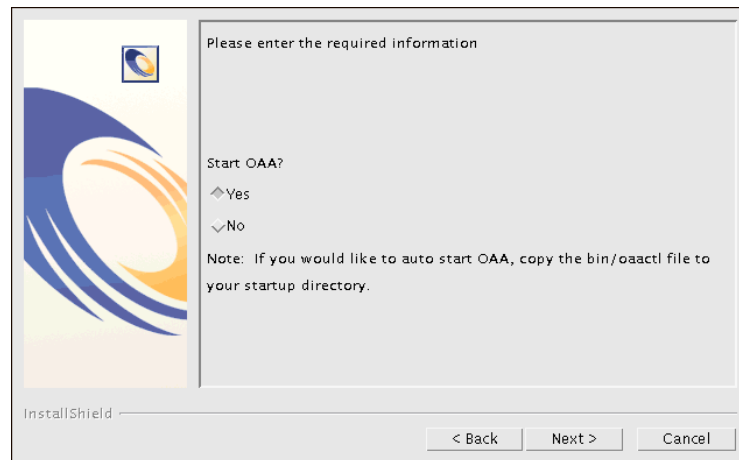
Click **Browse** and then select the **/usr/local** folder.

- **Figure 4.6. Typical installation in UNIX - Typical or Custom?**



Select the **Typical** option.

- **Figure 4.7. Typical installation in UNIX - Start OAA?**



Select **Yes** to start AssetCenter Web straight away.

Otherwise, select **No**.

If you want AssetCenter Web to start automatically each time the server is started, copy the **oaactl** file to the **startup** folder.

By default, this file is located in the `/usr/local/peregrine/bin` folder.
After executing this procedure, AssetCenter Web is fully installed in the standard fashion.

Custom installation

This section describes the custom installation of AssetCenter Web.

Packages of the Custom installation type

Compatible applications

For the list of supported packages: → [Supported environments](#) [page 19].

Communication ports

The communication ports used by the AssetCenter Web installation program depend on the packages you select.

Refer to the documentation of your Web server and your application server in order to determine which communication ports they require.

Table 4.6. Custom installation in UNIX - ports

Default port	Component that uses it	Alternate port
8005	Tomcat application server administration	8015
8009	Tomcat application server worker	8019
80	IIS Web Server	8081

After installation, you must configure AssetCenter Web to use alternate communication ports, in case the standard ports are already used.

Port conflicts

If you are using AssetCenter Web on a server using Oracle 9.2.0.1, you may encounter a port conflict on ports 8009 and 8080.

Refer to the documentation of your Web server or your application server to see whether they use these 2 ports.

If you are using Tomcat as application server with Oracle 9.2.0.1, then, by default, there will be a conflict with port 8009. To resolve this conflict, you have the following solutions available:

- Install Oracle 9.2.0.1 on a separate server (recommended).
- Configure Oracle so that it uses a different port.
- Configure the Tomcat instances so that they use a different port.

For more information on how to configure the Tomcat ports: → [Load balancing for application servers in Windows](#) [page 103].

For further information on resolving port conflicts: → [Checking for Tomcat port conflicts](#) [page 162].

Custom installation procedure



Note:

If you interrupt the installation before the end of the procedure, you must run the uninstallation program to delete the files that have been created.

To perform a custom installation of AssetCenter Web:

- 1 Log on to the server using an account that has **root** rights.
-



Important:

Make sure your temporary folder has at least 300 MB of free disk space. In Solaris, for example, the global temporary folder is **/tmp**.

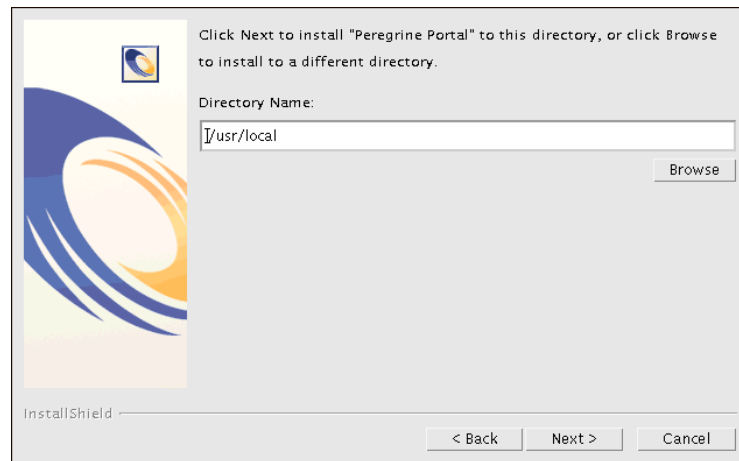
- 2 Insert the AssetCenter CD-ROM.
- 3 The installation program should start automatically. Exit the installation and then mount the CD-ROM directly.
Example: `mount /cdrom`
Go to the CD-ROM drive.
Example: `cd /cdrom`.
Enter the installation script corresponding to your operating system:

Table 4.7. Custom installation in UNIX - script to execute

Operating system	Command interpreter script to execute
AIX	./setupaix
Red Hat Linux	./setuplinux
Solaris	./setupsolaris

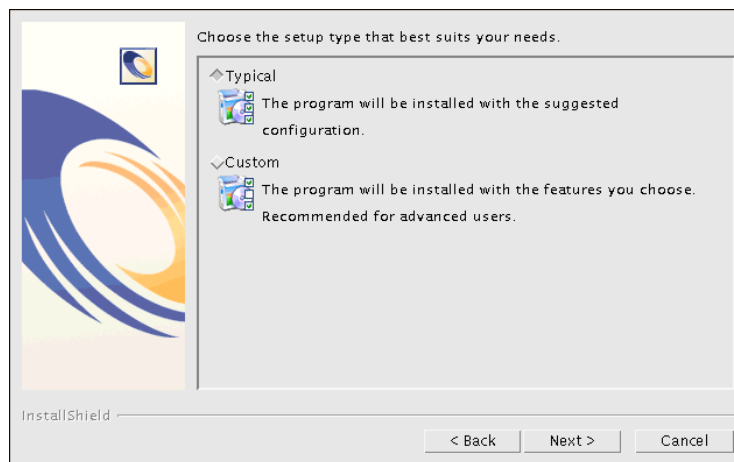
The welcome page of the installation program is displayed.

- 4 Here is some information about the screens that might raise questions:
 - **Figure 4.8. Custom installation in UNIX - Peregrine Portal folder**



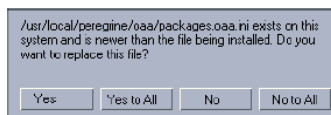
Click **Browse** and then select the **/usr/local** folder.

- **Figure 4.9. Custom installation in UNIX - Typical or Custom?**



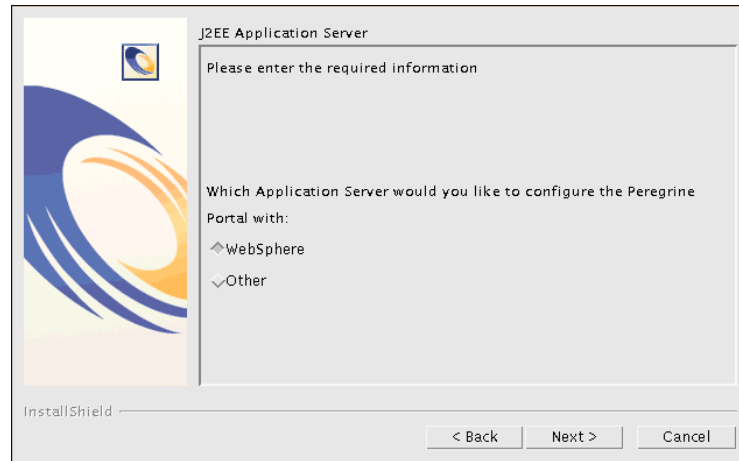
Select the **Custom** option.

- **Figure 4.10. Custom installation in UNIX - Replacing the packages.oaa.ini file**



Click **Yes to All**.

- **Figure 4.11. Custom installation in UNIX - Application server to configure**



Select **WebSphere** to configure a WebSphere application server or select **Other** to configure another application server.

- **Figure 4.12. Custom installation in UNIX - Folder of WebSphere application server**

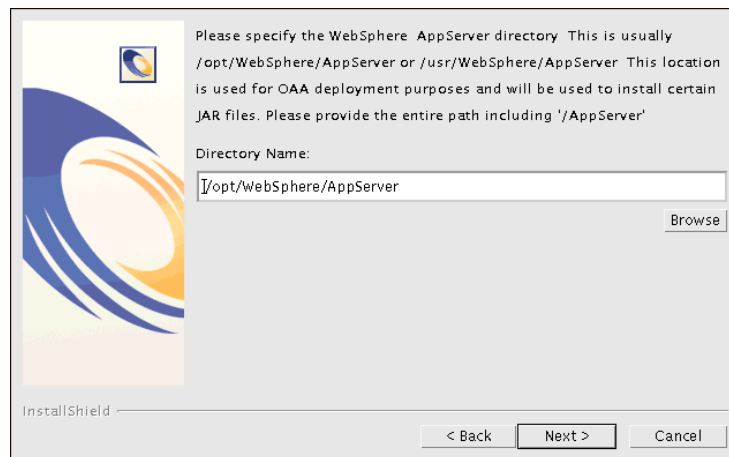
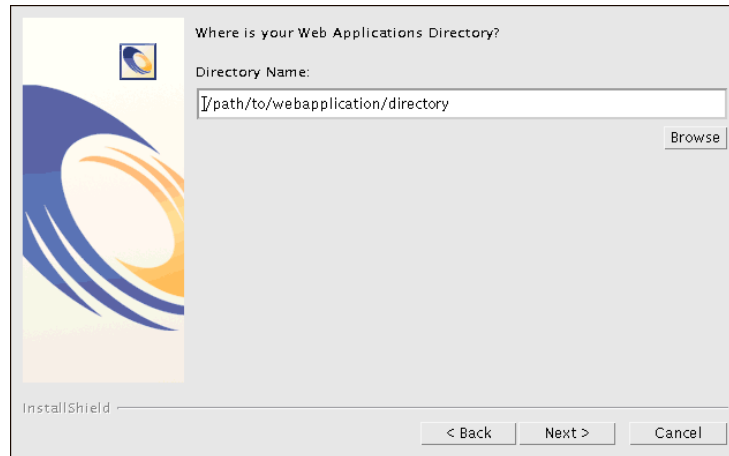
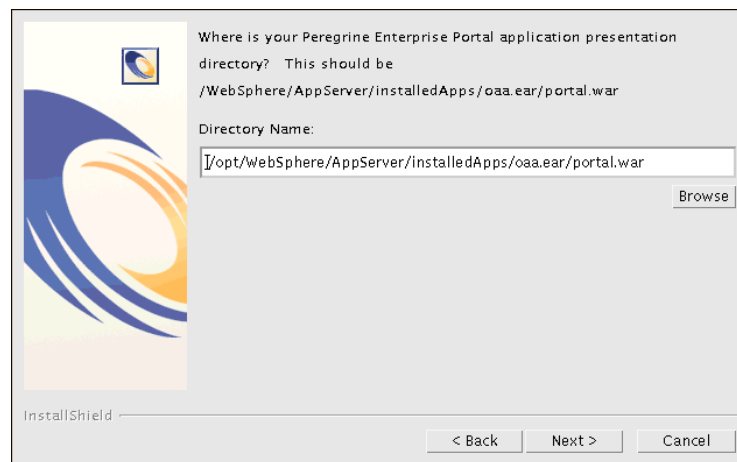


Figure 4.13. Custom installation in UNIX - Application server folder

Click **Browse** and then select the folder in which you have installed WebSphere AppServer or the alternate application server.

- **Figure 4.14. Custom installation in UNIX - Peregrine Enterprise Portal presentation folder**



Click **Browse** to select the folder in which you have deployed the **portal.<Version>.war** file.

For more information on how to deploy a **.war** file on the application server: → [Using an alternative application or Web server](#) [page 69].

- **Figure 4.15. Custom installation in UNIX - JDK folder used by WebSphere**

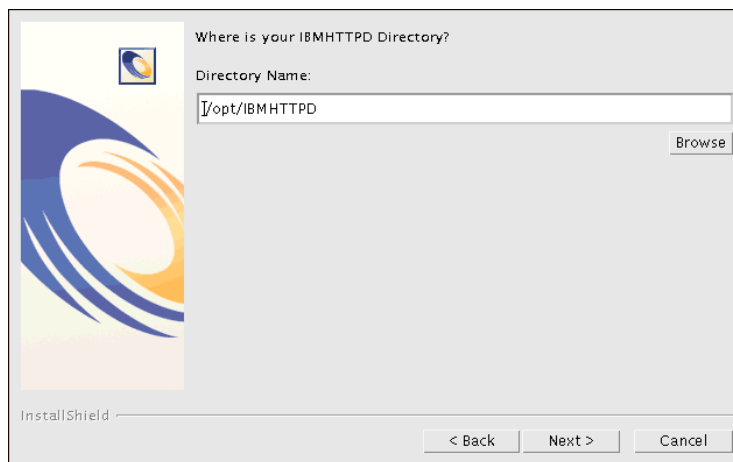


Figure 4.16. Custom installation folder in UNIX - JDK folder used by the application server



Click **Browse** to select the folder in which you have installed the Java Development Kit used by WebSphere or the alternate application server.

- **Figure 4.17. Custom installation folder in UNIX - IBM HTTPD folder**



Click **Browse** and select the folder in which you have installed IBM HTTP Server.

The installation program will automatically configure a virtual directory for the Web server called **oaa**.

- 5 Populate the environment variables concerning the DLL access paths for your operating system and application server.

Table 4.8. Custom installation in UNIX - DLL environment variables

Operating system	Environment variable	Paths to add
AIX	LIBPATH	<Application server>/WEB-INF/lib/AIX
Red Hat Linux 7.3	LD_LIBRARY_PATH	<Application server>/WEB-INF/lib/Linux
<ul style="list-style-type: none"> • Solaris 2.7 • Solaris 2.8 	LD_LIBRARY_PATH	<ul style="list-style-type: none"> • <Application server>/WEB-INF/lib/SunOS • /usr/ucblib

<App server>: Path of the context root of your application server, including the folder where you have installed the OAA application.

Use the following table to determine the context root of your application server:

Table 4.9. Custom installation folder in UNIX - Root context of the application server

Application server	Context root
WebSphere	/WebSphere/AppServer/installedApps/<AssetCenter Web folder>.ear /portal.4.0.0.<Version number of portal.<Version>.war file>/oaa
WebLogic	/bea/wlserver6.1/config/<WebLogic domain you have created>/applications/oaa

After executing this procedure, AssetCenter Web is fully installed and customized.

Configuring WebSphere in Linux or Solaris

If you are running Linux or Solaris, you can configure WebSphere using WebSphere Advanced Management Console.

To configure the WebSphere environment in Linux or Solaris:

- 1 Verify that the WebSphere Admin Server has been started.
- 2 Start WebSphere Advanced Administrator's Console:


```
/WebSphere/AppServer/bin/adminclient.sh
```
- 3 On the left of the console, click **Nodes/ <Server name>/ Application Servers**.
- 4 Click the application server on which you have installed AssetCenter Web.
- 5 Select the **General** tab.
- 6 Click **Environment**.
- 7 Click **Add**.
- 8 Enter the following information:
 - **Name:** **LD_LIBRARY_PATH**
 - **Value:** Enter the paths corresponding to your operating system. This path is made up of the access paths of the WebSphere applications concatenated with the relative access path of the **LIB** folders.
- 9 Click **OK**.

- 10 Click **Apply**.
- 11 Close WebSphere Advanced Administrator's Console.

Configuring the WebSphere startupServer.sh script in AIX

If you are running an AIX server, you must configure your WebSphere environment by editing the **startupServer.sh** script.

To configure the WebSphere startupServer.sh script in AIX:

- 1 Open the **startupServer.sh** file in a text editor.
- 2 Add a entry, **LIBPATH**, and set it to the path of AIX.

Example:

```
#!/bin/sh
LIBPATH=/usr/lib:/WebSphere/AppServer/installedApps/oa.ear/portal.4.0.0.55.war/WEB-INF/lib/AIX:/WebSphere/AppServer/installedApps/answer.ear/portal.4.0.0.55.war/WEB-INF/lib/AIX/ServiceCenter4
export LIBPATH
```

- 3 Save the file.

To install and configure ACAadapter in UNIX: → [Configuration](#) [page 131].

Uninstallation

This section describes the uninstallation procedure for AssetCenter Web.

Warning:

This procedure uninstalls all the components that you installed with the installation program.

If you selected the **Typical** installation, all components will automatically be uninstalled.

If you selected the **Custom** installation, only the components that you selected will be uninstalled.

To uninstall AssetCenter Web:

- 1 Open a command prompt.
- 2 Go to the folder <root>/usr/peregrine/_uninst

3



Note:

You must stop the servers before performing this step.

For this, enter the command:

```
./ooactl stop.
```

Enter the following command:

```
./uninstall.bin
```

- 4 Follow the instructions.

Tests

To verify that AssetCenter Web is correctly installed:

- 1 Check that the Web server and the application server are started.
- 2 Start a Web browser.
- 3 Enter the following URL:

```
http://<Name of the Web server>:<Port>/oaa/admin.jsp
```

The standard port number depends on the application server used:

Table 4.10. Installing in UNIX - Standard port numbers

Application server used	Standard port number
WebSphere	9080
WebLogic	7001
Tomcat	80, may be omitted from the URL

If everything is correctly installed and configured, the opening administration page is displayed.

If the opening page of the administration session does not display: →

Troubleshooting [page 155].

5 | Load balancing for application servers in Windows

CHAPTER

Load balancing on application servers

A server running a Web application such as AssetCenter Web uses about 256 MB of memory for each instance of the application server.

You must not set the JVM heap size beyond the size of the memory made available for the application server.

Exceeding the amount of available RAM causes the JVM processes to swap to disk, reducing overall performance. A setting of 256 MB should be sufficient for most systems.

Unlike other connectors, the AssetCenter creates a unique connection to its database.

For this reason, the memory consumed by the AssetCenter database server is the same as that used by the unique client connection.

Note that memory usage does not increase significantly per session, because the architecture is based on the sharing of a set of resources and database connections among all sessions handled by the same application server instance.

The small amount of memory consumed for session-specific information is released as the users log off or as their sessions expire.

Note that server sessions do not expire unless the browser is closed or the user navigates to a different domain.

Because the AssetCenter adapters maintain a single connection to the back end, adding extra application server instances brings the added benefit of concurrent access to the back-end data store.

The need for extra application server instances and therefore JVMs is directly related to three variables:

- The number of concurrent users.
- The processing power of the computer hosting the AssetCenter Web server.
- Number of processors on the computer.

Each deployment may make different demands of the software and hardware, but, in any case, optimal back-end throughput is achieved with the maximum number of application server instances that the server can handle without degraded performance due to lack of CPU headroom, file system swapping, and context switching.

Cache synchronization with Symmetric MultiProcessing (SMP) servers can, in most cases, be ignored as a performance tuning factor except in the case of the extremely large-scale systems.

To serve as a control guideline, low-end processors, such as a Pentium 450, should be capable of producing acceptable load handling for around 100 concurrent sessions on a single application server process.

A dual Pentium 1000 with 2 gigabytes of RAM (a common data center configuration) should be capable of handling 400+ concurrent sessions using multiple application server instances.

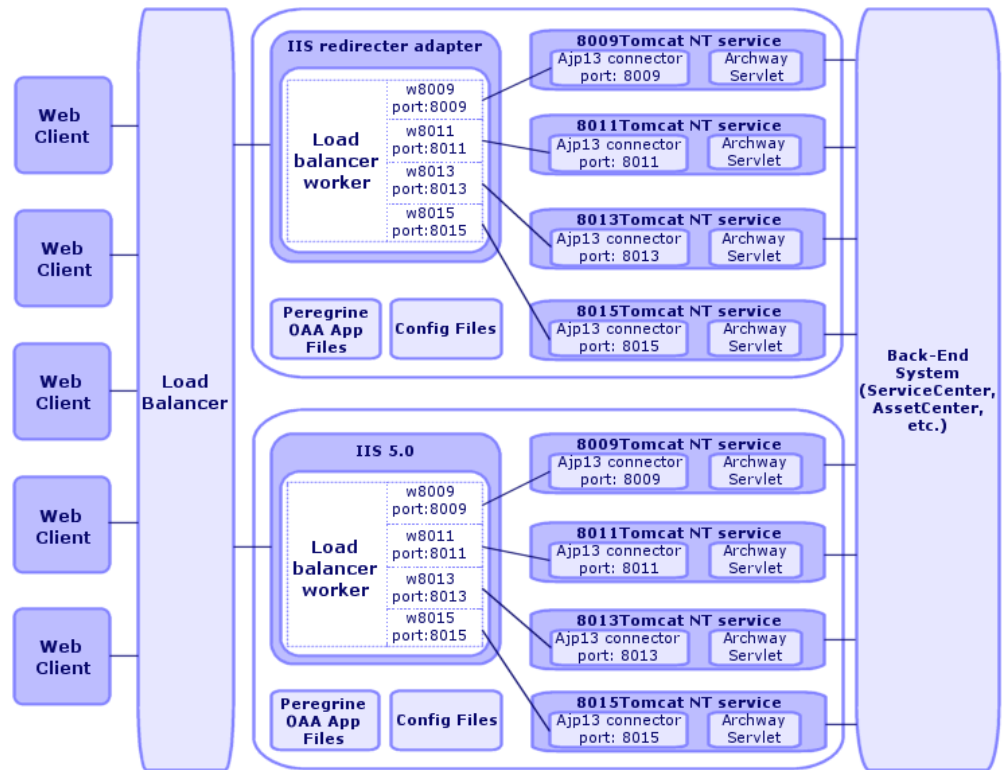
If you are using connector capable of pooling, such as JDBCAdapter or BizDocAdapter, performance beyond 400 concurrent users can be achieved.

The following diagram illustrates the architecture of multiple JVMs:

 **Note:**

A white paper on Peregrine OAA architecture and optimization is available on the Peregrine technical support Web site (<http://support.peregrine.com>).

Figure 5.1. Load balancing for application servers - architecture



Creating multiple instances of Tomcat for Apache

You can create multiple instances of Tomcat to balance the load of queries for AssetCenter Web. Each of the Tomcat instances can be configured as a service. Although this is not necessary, it increases performance, makes the instances easier to handle and adds additional functionality (restarting the server if it fails or if the computer on which the instances are installed needs to be restarted, for example).



Note:

The procedures given in this section assume that you have already installed AssetCenter Web.

If you use IIS: → [Creating multiple instances of Tomcat for IIS](#) [page 115].

To create multiple instance of Tomcat for Apache:

- 1 Disabling script pollers on the primary Tomcat instance [page 106]
- 2 Duplicate the Tomcat folder [page 107]
- 3 Editing the workers.properties file [page 107]
- 4 Editing the mod_jk.conf file [page 109]
- 5 Modifying the httpd.conf file [page 110]
- 6 Modifying the server.xml files for Apache [page 110]
- 7 Modifying the jk2.properties file for Apache [page 113]
- 8 Installing the Tomcat instances as Apache services [page 113]
- 9 Enabling script pollers on the primary Tomcat instance [page 114]
- 10 Testing load balancing in Apache [page 115]

Disabling script pollers on the primary Tomcat instance

You require one single instance of Tomcat to execute the scripts pollers.

Before copying the primary instance of Tomcat, connect to the AssetCenter Web administration page and disable script polling.

This will disable script polling for all Tomcat instances created by copying the primary instance.

- 1 Open an AssetCenter Web administration session.

The default URL of the administration page is:

```
http://<Name of the Web server>:<Port>/oaa/admin.jsp
```

- 2 Select the **Admin/ Settings** menu.
- 3 Locate the **Server-Side Scripts** section.
- 4 Set the **Enable script pollers** option to **No**.
- 5 Click the **Save** button at the bottom of the screen.
- 6 Click **Reset Server**.

- 7 End the AssetCenter Web. administration session.
 - 8 Stop the Tomcat service to disable AssetCenter Web temporarily.
- Disabling script pollers on the primary Tomcat instance:**

Duplicate the Tomcat folder

You must create a separate folder for each Tomcat instance used for load balancing.

To duplicate the Tomcat folder:

- 1 Open a Windows Explorer and copy the Tomcat installation folder.
By default, Tomcat is installed in the folder **C:\Program Files\Peregrine\Common\Tomcat4**.
- 2 Paste the copies in the same root folder.
Example:
C:\Program Files\Peregrine\Common
- 3 Rename the new folders.



Tip:

Include the port number that the instances will use in the name of the folder. For example, if you are going to use 4 instances of Tomcat listening to ports 8009, 8011, 8013 and 8015, create 3 copies of the Tomcat folder and name them **Tomcat4_8011**, **Tomcat4_8013** and **Tomcat4_8015**. The first instance will use port 8009.

- 4 Delete the **webapps\oaa** sub-folder from the copies of the Tomcat folder. The additional instances will use the same document root as the first or primary Tomcat instance.

Editing the workers.properties file

For each server on which the Tomcat instances are installed, only one **workers.properties** file is used.

Tomcat installs the **workers.properties** file in the **conf** folder of your primary Tomcat instance.

This file will be shared by all instances of Tomcat on the server.

The **workers.properties** file defines the worker threads that the Web server connector will create in order to communicate with the Tomcat instances.

Each Tomcat instance must communicate on a different port.

The host must have the name of the server running the Tomcat instances or **localhost** if the instances are running on the same server as Apache.

To modify the workers.properties file:

- 1 Open the **workers.properties** file in a text editor.

This file is located in the **conf** sub-folder of the Tomcat installation folder.

- 2 Modify the following lines as shown below.

The paths of the **workers.tomcat_home** and **workers.java.home** parameters match the Tomcat and Java SDK paths.

Example:

```
workers.tomcat_home="c:\Program Files\Peregrine\common\Tomcat4"
workers.java.home="c:\Program Files\Peregrine\common\jdk1.3.1_05"
ps=\
worker.list=loadbalancer, ajp13, w8011, w8013, w8015
```

Search for the string **worker.loadbalancer.type=lb** and modify the line following it as follows:

```
worker.loadbalancer.type=lb
worker.loadbalancer.balanced_workers=ajp13, w8011, w8013, w8015
```



Note:

You can define the worker names freely, as long as you use the same naming conventions throughout the procedure.

- 3 Add the following lines for each instance of Tomcat that you have installed, using the port numbers defined in the previous step.

```
worker.w8011.port=8011
worker.w8011.host=localhost
worker.w8011.type=ajp13
worker.w8011.cachesize=40
worker.w8011.lbfactor=10
```



Note:

All Tomcat instances share this **workers.properties** file; therefore, all additional lines must be in the file for the primary Tomcat instance.

- 4 Update the last 2 lines in the **Default ajp13 Worker Definition** section.

```
worker.ajp13.port=8009
worker.ajp13.host=localhost
worker.ajp13.type=ajp13
```

```
worker.ajp13.lbfactor=10           //Change lbfactor from =1 to =10
worker.ajp13.cachesize=40         //Change cachesize from =10 to =40
0
```

Search the string **worker.ajp13.lbfactor=10** and make the following modifications:

- 5 Save the file.

Editing the mod_jk.conf file

The **mod_jk.conf** file defines where the Worker files are available in Apache.

This file is shared by all Tomcat instances on the same server.

It is important that you do this procedure after you have successfully deployed the necessary AssetCenter Web files, otherwise the mount points, file locations and directories will not be included in the **mod_jk.conf** file and you will have to add them manually.

To edit the mod_jk.conf file:

- 1 Make a copy of the **mod_jk.conf** file and rename the copy to **mod_jk.conf-local**.

The **mod_jk.conf** file is located in the **Tomcat conf** directory.



Note:

This is only done on the primary Tomcat instance.

- 2 Open the **mod_jk.conf-local** file in a text editor.
- 3 Change **JKWorkersFile** to point to the **worker.properties** file of the primary Tomcat instance.

Example:

```
JkWorkersFile "C:\Program Files\Peregrine\Common\Tomcat4
\conf\worker.properties"
```

- 4 Change all **JkMounts** to use **loadbalancer** instead of **default worker ajp13**.

Syntax:

```
JkMount<file(s) or folder> <worker name>
```

Example:

```
JkMount/oaaservlet/* loadbalancer
JkMount/oaas/*.*.jsp loadbalancer
```

- 5 Save the file.

Modifying the httpd.conf file

The **httpd.conf** file must include **mod_jk.conf-local**.

To modify the httpd.conf file:

- 1 Open the **httpd.conf** file in a text editor.
This file is generally located in the **C:\Program Files\Peregrine\Common\Apache2\conf** folder.
- 2 Modify the following line in order to include the **mod_jk.conf-local** file:

```
include "<Tomcat installation folder>/conf/mod_jk.conf-local"
```

By default, Tomcat is installed in the folder **C:\Program Files\Peregrine\Common\Tomcat4**.

- 3 Save the file.

Modifying the server.xml files for Apache

You will need to modify the **server.xml** file for each Tomcat instance.

The **server.xml** file contains the information Tomcat needs to connect to the Web server as well as to find the Peregrine OAA Platform Web application files.



Tip:

Make a back up copy of the **server.xml** file before editing.

To modify the server.xml file:

Each Tomcat instance has a **server.xml** file located in the **conf** directory.

Modify the file for each instance of Tomcat:

- 1 Verify that the port number attribute of the **<Server>** element is a unique value that does not conflict with other port numbers used by Tomcat.
Example, when you are configuring 4 instances of Tomcat, we recommend using shutdown ports 8005 through 8008:

```
<Server port="8005" shutdown="SHUTDOWN" debug="0">
```

 Note:

These ports are not the worker communication ports (→ [Editing the workers.properties file](#) [page 107]).

- 2 Comment out a **<Connector>** tag with the **className="org.apache.coyote.tomcat4.CoyoteConnector"** parameter using port **8080**.

Tomcat uses this port to communicate with a browser for direct HTTP requests.

Since Apache will be serving the static data, Tomcat does not need to listen on this connector. It will also prevent a user from directly accessing Tomcat instances.

Example:

```
<!-- Define a non-SSL Coyote HTTP/1.1 Connector on port 8080 -->
<!--
<Connector className="org.apache.coyote.tomcat4.CoyoteConnector"
port="8080" minProcessors="5" maxProcessors="75"
enableLookups="true" redirectPort="8443" acceptCount="10" debug="0"
connectionTimeout="20000" useURValidationHack="false" />
-->
```

- 3 Change the port number used by the Coyote connector to a unique value that does not conflict with another port number.

If you configure 4 instances of Tomcat, the values 8009 (primary port), 8011, 8013 and 8015 are recommended .

Example:

```
<!-- Define a Coyote/JK2 AJP 1.3 Connector on port 8009 -->
<Connector className="org.apache.coyote.tomcat4.CoyoteConnector"
```

```
port="8009" minProcessors="5" maxProcessors="75"
enableLookups="true" redirectPort="8443" acceptCount="10" debug="0"
connectionTimeout="20000" useURIVValidationHack="false"
protocolHandlerClassName="org.apache.jk.server.JkCoyoteHandler" />
```

- 4 Update the **<Engine>** element with the server name and communications port used by each Tomcat instance:

```
<!-- Define the top level container in our container hierarchy -->
<Engine jvmRoute="localhost:8009" name="Standalone"
defaultHost="localhost" debug="0">
```

The port number should follow the convention used elsewhere in the configuration (8009, 8011, and so on).

These entries must be the same as the Tomcat ID entries you added to the **workers.properties** file in the primary Tomcat instance.

- 5 Update the **appBase** attribute of the **<Host>** element with the absolute path of the **webapps** directory of the primary Tomcat instance.

Example:

```
<!-- Define the default virtual host -->
<Host name="localhost" debug="0"
appBase="C:\Program Files\Peregrine\Common\Tomcat4\webapps"
unpackWARs="true" autoDeploy="true">
```

- 6 Create a **<Context>** element in the first instance or the primary instance of Tomcat and copy it to the other Tomcat instances, with the OAA context modified so it cannot be reloaded.

This prevents Tomcat from reloading the servlet without restarting the service.

It improves performance and helps keep the ISP code that the Tomcat instances are serving in sync during an update.

Add the entry just before the **Examples** entry.

Example:

```
<Context path="/oaa"
docBase="<First Tomcat install>/webapps/oaa"
crossContext="false"
debug="0"
reloadable="false" >
</Context>
```

For the **docBase** attribute, set **<First Tomcat install>** to the absolute path of the first or primary Tomcat instance.

- 7 Save the file.

Modifying the `jk2.properties` file for Apache

You will need to modify the `jk2.properties` file for each Tomcat instance. This file defines the JK2 communication port.

To modify the `jk2.properties` file:

For each instance of Tomcat:

- 1 Open the `jk2.properties` file in a text editor.
- 2 Add a line for the `channelSocket` port.
The port number must match the port number defined in the `workers.properties` file for this Tomcat instance.

Example:

```
channelSocket.port=8009
```

- 3 Save the file.

Installing the Tomcat instances as Apache services

After you have edited the Tomcat files, you can install each instance of Tomcat as Windows services using `installservice.bat`.

To install the Tomcat instances as Apache services:

For each Tomcat service to create:

- 1 Open a DOS command prompt and go to the Tomcat **bin** folder.
- 2 Enter the following command to create each instance of Tomcat:

```
installservice <Tomcat service name> <Installation folder of the Tomcat instance for which you have created the service> <Java SDK installation folder>
```

The second and third parameters are optional if you have already set the `CATALINA_HOME` and `JAVA_HOME` environment variables.

 **Warning:**

The command to create Tomcat instances does not support spaces in file paths.

Use the Windows naming conventions to avoid space problems in file paths. For example, replace **Program files** with **Progra~1**.

Example:

```
installservice Tomcat8009 C:\Progra~1\Peregrine\Common\Tomcat4_8009
C:\Progra~1\Peregrine\Common\jdk1.3.1_05\jre\bin\server\jvm.dll
```

 **Tip:**

You can easily remove a service. From the DOS command prompt, change directories to your Tomcat **bin** directory, then enter the following command:

```
Tomcat -uninstall <Service name>
```

The command is case-sensitive.

Enabling script pollers on the primary Tomcat instance

You require one single instance of Tomcat to execute the scripts pollers.

Before you test your load balancing configuration, you should login to the AssetCenter Web administration page of the primary Tomcat instance and turn on script script polling.

- 1 Open an AssetCenter Web administration session.

The default URL of the administration page is:

```
http://<Name of the Web server>:<Port>/oaa/admin.jsp
```

- 2 Select the **Admin/ Settings** menu.
- 3 Locate the **Server-Side Scripts** section.
- 4 Set the **Enable script pollers** option to **Yes**.
- 5 Click the **Save** button at the bottom of the screen.
- 6 Click **Reset Server**.
- 7 End the AssetCenter Web. administration session.

To enable script pollers on the primary Tomcat instance:

Testing load balancing in Apache

After creating the additional Tomcat instances, you can test whether the load is balanced using the following procedure.

To test the load balancing in Apache:

- 1 Start the services of all Tomcat instances.
If you have installed Tomcat as a service, you can open the Windows Control Panel and start each instance from the services dialog box.
 - 2 For each installed instance of Tomcat:
 - 1 Open a browser and then open an AssetCenter Web session.
 - 2 Execute an AssetCenter Web action, such as a search.
 - 3 Close the AssetCenter Web session.
 - 4 Close your browser to empty the connection caches.
-



Note:

If, for example, you have installed 4 instance of Tomcat, you must open and close a session 4 times.

The load balancing mechanism uses a **Round-Robin** algorithm.

If the load balancing is successful, each attempt to open a session will use a different instance of Tomcat.

- 3 Download the **archway.log** file.
You can download it from the AssetCenter Web administration portal, on the **Administration/ Server Log** page.
- 4 Open the **archway.log** file.
- 5 Make sure the connection information gives a different Tomcat instance for each connection.
If each session uses the same instance of Tomcat, the load is not correctly balanced and your must try to troubleshoot this.

Creating multiple instances of Tomcat for IIS

You can create multiple instances of Tomcat to balance the load of queries for AssetCenter Web. Each of the Tomcat instances can be configured as a service.

Although this is not necessary, it increases performance, makes the instances easier to handle and adds additional functionality (restarting the server if it fails or if the computer on which the instances are installed needs to be restarted, for example).



Note:

The procedures given in this section assume that you have already installed AssetCenter Web.

To create multiple instance of Tomcat for IIS:

- 1 Disabling script pollers on the primary Tomcat instance [page 116]
- 2 Duplicate the Tomcat folder [page 117]
- 3 Configuring the ISAPI plugin for IIS [page 118]
- 4 Creating and configuring a jakarta virtual directory in IIS [page 118]
- 5 Configuring IIS to use isapi_redirector2.dll as an ISAPI filter [page 119]
- 6 Creating and configuring a virtual directory oaa in IIS [page 120]
- 7 Modifying the workers2.properties file [page 121]
- 8 Modifying the server.xml files for IIS [page 122]
- 9 Modifying the jk2.properties file for IIS [page 124]
- 10 Installing the Tomcat instances as IIS services [page 125]
- 11 Enabling script pollers on the primary Tomcat instance [page 126]
- 12 Testing load balancing in IIS [page 126]

Disabling script pollers on the primary Tomcat instance

You require one single instance of Tomcat to execute the scripts pollers.

Before copying the primary instance of Tomcat, connect to the AssetCenter Web administration page and disable script polling.

This will disable script polling for all Tomcat instances created by copying the primary instance.

- 1 Open an AssetCenter Web administration session.

The default URL of the administration page is:

```
http://<Name of the Web server>:<Port>/oaa/admin.jsp
```

- 2 Select the **Admin/ Settings** menu.

- 3 Locate the **Server-Side Scripts** section.
 - 4 Set the **Enable script pollers** option to **No**.
 - 5 Click the **Save** button at the bottom of the screen.
 - 6 Click **Reset Server**.
 - 7 End the AssetCenter Web. administration session.
 - 8 Stop the Tomcat service to disable AssetCenter Web temporarily.
- Disabling script pollers on the primary Tomcat instance:**

Duplicate the Tomcat folder

You must create a separate folder for each Tomcat instance that you wish to use for load balancing.

To duplicate the Tomcat folder:

- 1 Open a Windows Explorer and copy the Tomcat installation folder.
By default, Tomcat is installed in the folder **C:\Program Files\Peregrine\Common\Tomcat4**.
- 2 Paste the copies in the same root folder.
Example:
C:\Program Files\Peregrine\Common
- 3 Rename the new folders.



Include the port number that the instances will use in the name of the folder. For example, if you are going to use 4 instances of Tomcat listening to ports 8009, 8011, 8013 and 8015, create 3 copies of the Tomcat folder and name them **Tomcat4_8011**, **Tomcat4_8013** and **Tomcat4_8015**. The first instance will use port 8009.

- 4 Delete the **webapps\oaa** sub-folder from the copies of the Tomcat folder. The additional instances will use the same document root as the first or primary Tomcat instance.

Configuring the ISAPI plugin for IIS

The AssetCenter Web installation program automatically places a copy of the ISAPI plugin for IIS in the following folder:

c:\Program Files\Peregrine\Common\Tomcat4\bin

To configure the ISAPI plugin for IIS for your Intranet environment:

- 1 Open the **jk2.reg** file in a text editor.

By default, this file is located in:

```
C:\Program Files\Peregrine\Common\Tomcat4\conf
```

- 2 Make sure the **serverRoot** and **workersFile** values correctly reference the Tomcat installation folder. By default, these values are:

```
"ServerRoot"="C:\\Program Files\\Peregrine\\Common\\Tomcat4"
"workersFile"="C:\\Program Files\\Peregrine\\Common\\Tomcat4\\conf\\
workers2.properties"
```

- 3 Save and close the **jk2.reg** file.
- 4 In the Windows explorer, double-click the **jk2.reg** file.
This modifies the Registry.

Creating and configuring a jakarta virtual directory in IIS

The ISAPI plugin for IIS requires a specific virtual directory in order to run. Use the following guidelines to create the virtual directory on the Default Web Site.

For specific instructions about configuring IIS, refer to your Windows Help.

To configure a jakarta virtual directory in IIS:

- 1 Start the Windows Internet Services management console (**Start/ Settings/ Control Panel/ Administrative Tools/ Internet Services (IIS)** menu in Windows XP).
- 2 Right-click the **Default Web Site** node and select the **New/ Virtual directory** menu.
- 3 Create the virtual directory using the following parameters:

Table 5.1. Multiple instances of Tomcat for IIS - Creating a virtual directory jakarta

Specification	Value
Virtual directory alias	jakarta
Web site content directory	<Tomcat installation folder>\bin The default installation folder of Tomcat is: C:\Program Files\Peregrine\Common\Tomcat4 This folder must contain the isapi_redirector2.dll file.
Access authorizations	Read, Execute scripts, Execute

Configuring IIS to use isapi_redirector2.dll as an ISAPI filter

To establish a connection between Tomcat and IIS, you will need to install the file **isapi_redirector2.dll** as an ISAPI filter.

To configure isapi_redirector2.dll as an ISAPI filter:

- 1 Start the Windows Internet Services management console (**Start/ Settings/ Control Panel/ Administrative Tools/ Internet Services (IIS)** in Windows XP).
- 2 Right-click the **Default Web Site** node and select the **Properties** menu.
- 3 Select the **ISAPI Filters** tab.
- 4 Click **Add**.
- 5 Set the **Filter Name** field to **jakarta**.
The filter name must match the name defined in the **jk2.reg** file. By default, this name is **jakarta**.
→ [Configuring the ISAPI plugin for IIS \[page 118\]](#)
- 6 Populate the **Executable** field with the value **isapi_redirector2.dll**.
By default, this file is located in **C:\Program Files\Peregrine\Common\Tomcat4\bin**.
- 7 Click **OK**.
- 8 Stop and restart the **IIS** server.
- 9 Restart the **PeregrineTomcat** service.

- 10 Start the Windows Internet Services management console (**Start/ Settings/ Control Panel/ Administrative Tools/ Internet Services (IIS)** in Windows XP).
- 11 Right-click the **Default Web Site** node and select the **Properties** menu.
- 12 Select the **ISAPI Filters** tab.
- 13 Verify that the service is running (green status arrow).
- 14 Exit the Internet Services management console.

Creating and configuring a virtual directory oaa in IIS

To execute AssetCenter Web with IIS, you must create a virtual directory that maps the Tomcat installation folder.

For specific instructions about configuring IIS, refer to your Windows Help.

To configure a oaa virtual directory in IIS:

- 1 Start the Windows Internet Services management console (**Start/ Settings/ Control Panel/ Administrative Tools/ Internet Services (IIS)** menu in Windows XP).
- 2 Right-click the **Default Web Site** node and select the **New/ Virtual directory** menu.
- 3 Create the virtual directory using the following parameters:

Table 5.2. Multiple instances of Tomcat for IIS - Creating a virtual directory oaa

Specification	Value
Virtual directory alias	oaa
Web site content directory	<Tomcat installation folder>\webapps\oaa The default installation folder of Tomcat is: C:\Program Files\Peregrine\Common\Tomcat4 This folder must contain the isapi_redirector2.dll file.
Access authorizations	Read, Execute scripts

<Tomcat installation folder>\webapps\oaa: We recommend using the name **oaa**.

If you chose another name, you must replace **oaa** everywhere else it is referenced:

- **<Tomcat installation folder>\webapps\oaa** folder.
- [uri] mappings in the **workers2.properties** file.
- **oaa** entries in the **mod_jk2.conf** file.
- **<Context>** path and **docBase** attributes of the **server.xml** file.

Important:

The other name must appear in the access URL of AssetCenter Web.

Example: `http://<Server name>/<Name of the virtual directory>/login.jsp`

Modifying the workers2.properties file

For each server on which Tomcat instances are installed, there is only one **workers2.properties** file.

Tomcat installs the **workers2.properties** file in the **conf** folder of your primary Tomcat instance.

This file will be shared by all instances of Tomcat on the server.

The **workers2.properties** file defines the worker threads that the Web server connector will create in order to communicate with the Tomcat instances.

Each Tomcat instance must communicate on a different port.

The host must have the name of the server running the Tomcat instances or **localhost** if the instances are running on the same server as IIS.

To modify the workers2.properties file:

- 1 Open the **workers2.properties** file in a text editor.
This file is located in the **conf** sub-folder of the Tomcat installation folder.
- 2 Create an entry, **channel.socket** for each Tomcat process (also called **worker**).

Example:

```
[channel.socket:<server>:<port>]
info=Description of the Tomcat instance
debug=0
tomcatId=<Tomcat instance server name>:<Listening port>
```

```
lb_factor=1
disabled=0
```

The **lb_factor** is a number greater than or equal to 1 that IIS uses to load balance the workers.

If all the workers are running on servers that have equal performance strengths, you should set the **lb_factor** numbers to equal values (typically 1).

If you want to assign fewer user sessions to a given Worker, then assign it a lower **lb_factor** number relative to the other Workers.

- 3 Verify that the **uri** settings lists the proper IIS virtual directory.

By default, this virtual directory is **oaa**.

If you defined a different virtual directory other than **oaa** to run AssetCenter Web you will need to change the **uri** values here.

Example:

```
uri:/oaa/servlet/*]
info=Prefix mapping
[uri:/oaa/*.jsp]
info=Extension mapping
```

- 4 Save the file.

Modifying the server.xml files for IIS

You will need to modify the **server.xml** file for each Tomcat instance.

The **server.xml** file contains the information Tomcat needs to connect to the Web server as well as to find the Peregrine OAA Platform Web application files.



Tip:

Make a back up copy of the **server.xml** file before editing.

To modify the server.xml file:

Each Tomcat instance has a **server.xml** file located in the **conf** directory.

Open this file in a text editor for each instance of Tomcat:

- 1 Verify that the port number attribute of the **<Server>** element is a unique value that does not conflict with other port numbers used by Tomcat.

Example, when you are configuring 4 instances of Tomcat, we recommend using shutdown ports 8005 through 8008:

```
<Server port="8005" shutdown="SHUTDOWN" debug="0">
```

 **Note:**

These ports are not the worker communication ports (→ [Editing the workers.properties file](#) [page 107]).

- 2 Change the port number used by the Coyote connector to a unique value that does not conflict with another port number.

If you configure 4 instances of Tomcat, the values 8009 (primary port), 8011, 8013 and 8015 are recommended.

Example:

```
<Connector className="org.apache.coyote.tomcat4.CoyoteConnector"
port="8009" minProcessors="5" maxProcessors="75" enableLookups="true"
redirectPort="8443" acceptCount="10" debug="0"
connectionTimeout="20000" useURIVValidationHack="false"
protocolHandlerClassName="org.apache.jk.server.JkCoyoteHandler" />
```

- 3 Create a **<Context>** element in the first instance or the primary instance of Tomcat and copy it to the other Tomcat instances, with the OAA context modified so it cannot be reloaded.

This prevents Tomcat from reloading the servlet without restarting the service.

It improves performance and helps keep the ISP code that the Tomcat instances are serving in sync during an update.

Add the entry just before the **Examples** entry.

Example:

```
<Context path="/oaa"
docBase="<First Tomcat install>/webapps/oaa"
crossContext="false"
debug="0"
reloadable="false" >
</Context>
```

For the **docBase** attribute, set **<First Tomcat install>** to the absolute path of the first or primary Tomcat instance.

- 4 Update the **jvmRoute** attribute of the **<Engine>** element with the server name and communications port used by each Tomcat instance.

Example:

```
<Engine jvmRoute="localhost:8009" name="Standalone"
defaultHost="localhost" debug="0">
```

- 5 Update the **<Host>** element with the **webapps** directory used by the first or primary Tomcat instance.

List the server information in the **appBase** attribute.

Example:

```
<Host name="localhost" debug="0"
appBase="<Absolute path of the first instance or primary instance o
f Tomcat>/webapps" unpackWARs="true"
autoDeploy="true">
```

- 6 Comment out a **<Connector>** tag with the **className="org.apache.coyote.tomcat4.CoyoteConnector"** parameter using port **8080**.

Tomcat uses this port to communicate with a browser for direct HTTP requests.

```
Example:
<!-- Define a non-SSL Coyote HTTP/1.1 Connector on port 8080 -->
<Connector className="org.apache.coyote.tomcat4.CoyoteConnector"
port="8080" minProcessors="5" maxProcessors="75"
acceptCount="10" debug="0" connectionTimeout="20000"
useURValidationHack="false" />
-->
```

- 7 Save the file.

Modifying the **jk2.properties** file for IIS

You will need to modify the **jk2.properties** file for each Tomcat instance.

This file defines the JK2 communication port.

To modify the **jk2.properties** file:

For each instance of Tomcat:

- 1 Open the **jk2.properties** file in a text editor.
- 2 Add a line for the **channelSocket** port.

The port number must match the port number defined in the **workers2.properties** file for this Tomcat instance.

Example:

```
channelSocket.port=8009
```

→ [Modifying the workers2.properties file](#) [page 121]

- 3 Save the file.

Installing the Tomcat instances as IIS services

After you have edited the Tomcat files, you can install each instance of Tomcat as Windows services using **installservice.bat**.

To install the Tomcat instances as IIS services:

For each Tomcat service to create:

- 1 Open a DOS command prompt and go to the Tomcat **bin** folder.
- 2 Enter the following command to create each instance of Tomcat:

```
installservice <Tomcat service name> <Installation folder of the Tomcat instance for which you have created the service> <Java SDK installation folder>
```

The second and third parameters are optional if you have already set the **CATALINA_HOME** and **JAVA_HOME** environment variables.

Warning:

The command to create Tomcat instances does not support spaces in file paths.

Use the Windows naming conventions to avoid space problems in file paths. For example, replace **Program files** with **Progra~1**.

Example:

```
installservice Tomcat8009 C:\Progra~1\Peregrine\Common\Tomcat4
C:\Progra~1\Peregrine\Common\jdk1.3.1_05\jre\bin\server\jvm.dll
```

- 3 Start the Tomcat instance you have just created.

Tip:

You can easily remove a service. From the DOS command prompt, change directories to your Tomcat **bin** directory, then enter the following command:

```
Tomcat -Uninstall <Service name>
```

Enabling script pollers on the primary Tomcat instance

You require one single instance of Tomcat to execute the scripts pollers.

Before you test your load balancing configuration, you should login to the AssetCenter Web administration page of the primary Tomcat instance and turn on script script polling.

- 1 Open an AssetCenter Web administration session.

The default URL of the administration page is:

```
http://<Name of the Web server>:<Port>/oaa/admin.jsp
```

- 2 Select the **Admin/ Settings** menu.
- 3 Locate the **Server-Side Scripts** section.
- 4 Set the **Enable script pollers** option to **Yes**.
- 5 Click the **Save** button at the bottom of the screen.
- 6 Click **Reset Server**.
- 7 End the AssetCenter Web. administration session.

To enable script pollers on the primary Tomcat instance:

Testing load balancing in IIS

After creating the additional Tomcat instances, you can test whether the load is balanced using the following procedure.

To test the load balancing in IIS:

- 1 Start the services of all Tomcat instances.
If you have installed Tomcat as a service, you can open the Windows Control Panel and start each instance from the services dialog box.
- 2 For each installed instance of Tomcat:
 - 1 Open a browser and then open an AssetCenter Web session.
 - 2 Execute an AssetCenter Web action, such as a search.
 - 3 Close the AssetCenter Web session.
 - 4 Close your browser to empty the connection caches.



If, for example, you have installed 4 instance of Tomcat, you must open and close a session 4 times.

The load balancing mechanism uses a **Round-Robin** algorithm.

If the load balancing is successful, each attempt to open a session will use a different instance of Tomcat.

-
- 3 Download the **archway.log** file.

You can download it from the AssetCenter Web administration portal, on the **Administration/ Server Log** page.

- 4 Open the **archway.log** file.
- 5 Make sure the connection information gives a different Tomcat instance for each connection.

If each session uses the same instance of Tomcat, the load is not correctly balanced and your must try to troubleshoot this.

6 | Implementing the servers

CHAPTER

To configure OAA and make AssetCenter Web available to users, you must make sure that the following applications are always running on the server:

- The application server (Tomcat by default).
- The Web server (Apache by default).

If everything goes according to plan, the installation program automatically starts Tomcat and Apache when it finishes its execution.

If also makes sure that Tomcat and Apache start automatically each time the computer is started.

This chapter explains how to start Tomcat and Apache if they are stopped.

Starting Tomcat

To stop or start Tomcat:

- 1 Open a DOS command prompt.
- 2 Go to the **common\Tomcat4\bin** sub-folder of the AssetCenter Web installation folder.
- 3 Enter the following command:

```
startup
```

- 4 Leave the command prompt open.



Note:


In Unix, the command is:

```
tomcat.sh start
```

For more information about Tomcat, refer to its documentation.

Starting Apache

To start Apache:

- 1 Start a Windows browser.
- 2 Select the **common\Apache2\bin** sub-folder in the Apache installation folder.
- 3 Execute **ApacheMonitor.exe**.
- 4 Right-click the  icon on the Tasks toolbar.
- 5 Select **Open Apache Monitor**.
- 6 Click **Start**.

For more information about Apache, refer to its documentation.

7 Configuration

CHAPTER

Before a user can connect to AssetCenter Web you must configure OAA using the administration portal.

This chapter explains how to do that.

Connecting to the administration portal

To connect to the administration portal:

- 1 Verify that the application server (Tomcat by default) and the Web server (Apache by default) are started on the AssetCenter Web server.



It is not necessary to start AssetCenter.

- 2 Start a Web browser.
- 3 In the browser's **Address** field, type:

```
http://<host>:<port>/oaa/admin.jsp
```

host: name of the AssetCenter Web server.

- port:** shows the port number if it is different from **80**.
- 4 Press **ENTER**.
The administration module connection page is displayed.
 - 5 In the **User Name** field, enter **System**.
The password is empty by default.
 - 6 Click the **System Maintenance login** button.

Administration menu

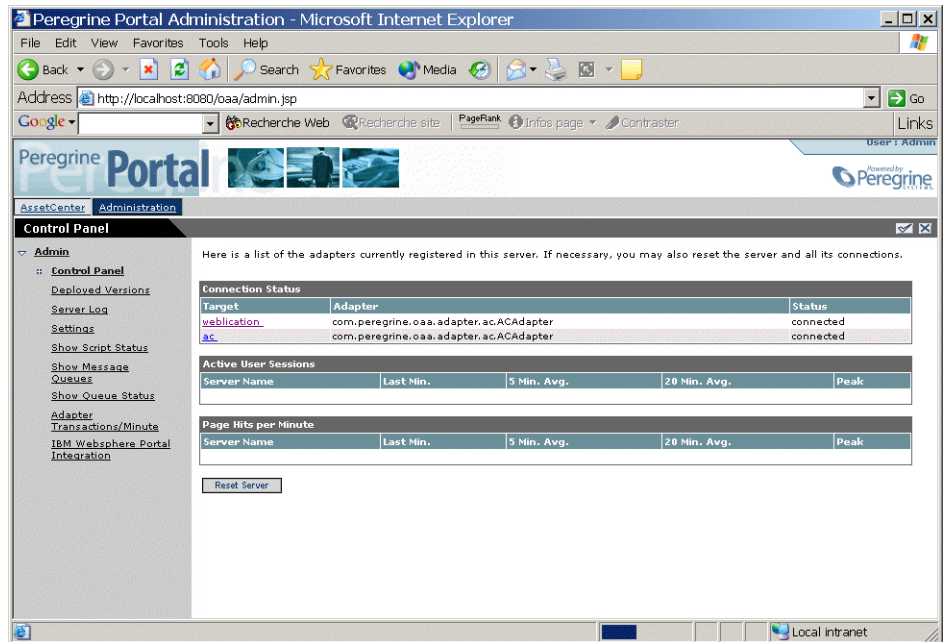
The **Admin** menu on the left gives you access to the different administration modules.

Control Panel

To configure the Control Panel module:

- 1 Select the **Control Panel** menu.

Figure 7.1. Administration portal - Control Panel



- 2 Verify that the following connectors appear in the list (if there are others in addition it is not important):

- **weblication**
- **ac**



Note:

At this stage, it is possible that these connectors have the **Disconnected** status. This is not a problem.

AssetCenter connector

For a Windows server

To configure the AssetCenter connector:

- 1 Select the **Settings** menu.
- 2 Select the **AssetCenter** tab.
- 3 Verify the parameters on this page, in particular:
 - **Database** : name of the AssetCenter connection to the database.



Note:

The AssetCenter must already be declared on the AssetCenter Web server using an AssetCenter client.

- 4 Click **Save** to save any modifications.

For a UNIX server

To configure the AssetCenter connector:

- 1 Install an AssetCenter client on the UNIX server as set out in the **AssetCenter 4.3 - Installation Guide**, chapter **Installing and configuring in UNIX**, section **Installing AssetCenter**.
- 2 Select the **Settings** menu.
- 3 Select the **AssetCenter** tab.
- 4 Verify the parameters on this page, in particular:
 - **Database**: Name of the AssetCenter database connection as defined in the **amdb.ini** file you have placed on the server.
Location of this file: → **AssetCenter - Installation** guide, chapter **.ini and .cfg files**.
 - **AC Shared Library Name**: Name of the DLL in the **AssetCenter/bin** directory.
Example: **aamapi43** for **libaamapi43.so**.

Java automatically adds the prefix and suffix when it searches for the file.

- **AC Shared Library Path:** Full path of the **bin** sub-folder of the AssetCenter installation folder; a slash / must be included at the end of the path.

5 Click **Save** to save any modifications.

Available languages

The user can select the language for the AssetCenter Web interface in an itemized list of supported languages on the connection page.

To define the supported languages and the default language:

- 1 Select the **Settings** menu.
- 2 Select the **Common** tab.
- 3 In the **Encoding, Locales, and Sessions**, select the parameter **Locales**.
- 4 Define the available languages for the user interface of AssetCenter Web.

Example:

The first value contained in the **Locales** parameter determines the **default language**.

The following list presents the available languages with their corresponding country codes:

Table 7.1. Settings - Languages

Language	Language code
English	en
French	fr

5 Click **Save** to save any modifications.

User rights

There are several types of rights:

Table 7.2. Settings - Defining user rights

Types of rights	Defining the types of rights
AssetCenter Web pages visible to all users	<ol style="list-style-type: none"> 1 Start the AssetCenter Web administration module with the System login. <div style="border: 1px solid black; padding: 2px; margin: 5px 0;"><code>http://<host>:<port>/oaa/admin.jsp</code></div> 2 Select the Settings menu. 3 Select the AssetCenter tab. 4 Select the Default Capability Words parameter. 5 Indicate the list of rights. Syntax: <div style="border: 1px solid black; padding: 2px; margin: 5px 0;"><code>ac(<right 1>;<right 2>;...;<right n>)</code></div> Example: <div style="border: 1px solid black; padding: 2px; margin: 5px 0;"><code>ac(acw.portfolio;acw.contract;acw.finance;acw.system)</code></div> <p>To learn the list of rights, refer to the section List of AssetCenter Web access rights [page 138].</p> <p>Note:</p> <p>Rights grant access to the pages accessible from the main menu or using the links to these pages (from whatever page on which these links are located).</p>

Types of rights**Defining the types of rights**

AssetCenter Web visible by a given user.

- 1 Start AssetCenter.
- 2 Display the list of user rights via the **Administration/ User rights** menu.
- 3 If they do not exist, create the **acw.*** rights.
To learn the list of available rights: → [List of AssetCenter Web access rights](#) [page 138].
At this stage, only the name of the rights counts; you can define the read rights, entry rights creation rights and destruction rights later.
- 4 Close the list of user rights by clicking **Close**.
- 5 Display the list of user profiles via the **Administration/ User profiles** menu.
- 6 Create a new profile or select an existing one.
- 7 Select the **User rights** tab.
- 8 Add the appropriate **acw.*** rights.
- 9 Save this information via the **Create** or **Modify** buttons.
- 10 Close the list of user profiles by clicking **Close**.
- 11 Display the list of employees via the **Portfolio/ Departments and Employees** menu.
- 12 Select the person to whom you want to assign the access profile you just created.
- 13 Select the **Profile** tab.
- 14 Populate the **Profile** field by selecting the access profile that you just defined.
- 15 Close the list of employees by clicking **Close**.

If the **acw.*** rights are not in your AssetCenter database (if it originates from a version prior to 4.3), you can import the user rights and the standard access profiles.

For this:

- 1 Start AssetCenter.
- 2 Connect to your database with administrative rights.
- 3 Select **File/ Import/ Execute a script**.
- 4 Select the script **profile.scr**.

By default, this script is located in the folder **C:\Program Files\Peregrine\AssetCenter\datakit\acw**.

- 5 Click **Import**.


The names of the AssetCenter Web profiles are prefixed with **AssetCenter Web profile**.

Types of rights	Defining the types of rights
AssetCenter database data visible and modifiable by a given user.	1 Start AssetCenter.
	2 Display the list of employees via the Portfolio/ Departments and Employees menu.
	3 Select the employee to whom you want to give access to the AssetCenter database.
	4 Select the Profile tab.
	5 Display the detail of the profile associated to the employee by clicking the magnifying glass to the right of the Profile field.
	6 Define the profile completely (particularly the user rights and the access restrictions). For the access rights, you must define the read rights, entry rights, creation rights and destruction rights. You can use the general rights or the acw.* rights that were designed for the display of the AssetCenter Web pages.
	7 Save these modifications by clicking Modify .

List of AssetCenter Web access rights

Table 7.3. Settings - List of access rights

Name of the access right	Corresponding module	Authorized pages
acw.cable	Cable	All pages of the module.
acw.contract	Contracts	All pages of the module.
acw.finance	Finance	All pages of the module.
acw.portfolio	Portfolio	<ul style="list-style-type: none"> • Portfolio elements • Batches • Computers • Telephones • Projects
acw.portfolio + acw.portfolio.technician	Portfolio items	Pages authorized by the acw.portfolio right, plus: <ul style="list-style-type: none"> • My work orders
acw.portfolio + acw.portfolio.manager	Portfolio item	Pages authorized by the acw.portfolio right, plus: <ul style="list-style-type: none"> • Work orders

Name of the access right	Corresponding module	Authorized pages
acw.system	General	<ul style="list-style-type: none"> • Companies • Contacts • Departments • Employees • Groups • Locations • Stocks • My Messages
acw.system + acw.system.mytasks	General	Pages authorized by the acw.system right, plus: <ul style="list-style-type: none"> • My tasks in progress
acw.reconc	Reconciliation	Reconciliation proposals assigned to the assignee connected to AssetCenter Web.
acw.reconc + acw.reconc.manager	Reconciliation	All reconciliation proposals.
getit.developer		<p>This right is only taken into account if the End user personalization option (Common tab), section Personalization is set to Disabled.</p> <p>In this case, this right enables customization of all parameters of pages in AssetCenter Web using the  icon.</p>

Defining the AssetCenter Web personalization rights

The ability to customize the AssetCenter Web pages depends on:

- The value of the **End user personalization** option (**Common** tab, section **Personalization**)
- The assignment of the **getit.developer** user right in the **Default Capability Words** (**AssetCenter** tab)
- The definition and the assignment of the **getit.developer** user right in the AssetCenter database.

To define the value of the Personalization option:

- 1 Select the **Settings** menu.
- 2 Select the **Common** tab.
- 3 In the **Personalization** section, select the parameter **End user personalization**.
- 4 Assign one of the following values:
 - **Enabled**: All users can customize all aspects of the AssetCenter Web pages to which they have access.
 - **Disabled** - only the users with a profile with the **getit.developer** right can customize all aspects of the AssetCenter Web pages to which they have access.
 - **Limited** - All users can customized certain aspects of the AssetCenter Web pages to which they have access.
- 5 Click **Save** to save any modifications.

Validating new parameters

Click the **Control panel** menu and then click **Reset Server**.

At this stage, the **ac** and **weblication** must have the status **Connected**.

Modifying a parameter

To define or modify a parameter:

- 1 Select the **Settings** menu.
- 2 Select the tab containing the parameter to modify.
- 3 Enter the new value of the parameter.



It is sometimes possible to revert back to a parameter's default value by clicking on the link located to the right of the parameter.

-
- 4 Click **Save** on the bottom of the page.

- 5 After the parameters are modified and saved, click **Reset Server**. When the server finishes refreshing, a message appears on the top of the **Control Panel** page.

8 Utilization

CHAPTER

This chapter provides information on:

- How to use AssetCenter Web.
- The capabilities of the standard AssetCenter Web Web applications.

AssetCenter demonstration database

If you have selected this option, AssetCenter is installed with a demonstration database.

This demonstration database includes data spanning most tables and is intended to help you in testing AssetCenter Web.

The user **Richard Hartke** is associated with the **Full AssetCenter Web profile**, thus giving him access to all AssetCenter Web pages and the corresponding data in the AssetCenter database.

His login is **Hartke**; The associated password is empty.

General interface

Connecting to AssetCenter Web

To connect to AssetCenter Web:

- 1 Enter the following URL in your browser.

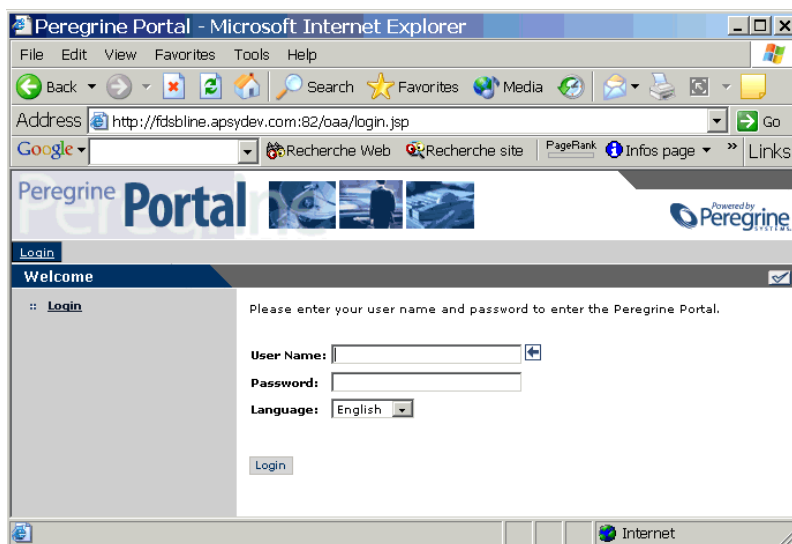
```
http://<host>:<port>/oaa/login.jsp
```

host: name of the AssetCenter Web server.

port: The port number if it is different from **80**.

- 2 The AssetCenter Web home page appears in your browser.

Figure 8.1. AssetCenter Web - Home page



- 3 Populate the fields and then click **Login**.



Tip:
To connect to the demonstration database: → [AssetCenter demonstration database](#) [page 143].

Tabs

Several tabs are displayed in the top part of the screen:

- **Home:** Enables customization of the home page and certain general parameters at the user computer level.
- **AssetCenter:** Enables access to the screens in the AssetCenter modules.

Main menu

The sidebar of the AssetCenter Web page contains a list of all the available Web applications depending on the tab activated in the top part of the screen. Click a menu item to access the desired Web application.

Browsing window




The right-hand side of the AssetCenter Web page is the information browser window. In this window are displayed the results of your searches and the modifications made to records.

Toolbar

The table below lists the different toolbar icons and the functions that they represent.

Table 8.1. Interface - Toolbar icons




Icon	Function
	This icon ends your AssetCenter Web work session.

Icon	Function
	This icon enables you to trigger an action that depends on the context in which the icon is displayed. The ToolTip that is displayed when the pointer is left over the icon tells you exactly which action is triggered.
	This icon enables you to customize the current page at the user computer level.
	This icon enables you to automatically translate data from the AssetCenter database.


Icons

You will find the following icons in the Web applications:

Table 8.2. Interface - Page icons

Icon	Function
	Displays the list of records among which you will make your choice.
	Deletes the contents of a field.
	Displays a graphic calendar that enables you to select a date.

Disconnect

To disconnect, simply click  in the AssetCenter Web toolbar.

Default Web applications

The standard version of AssetCenter Web comes with a set of simple functions. To access these functions:

- Connect to the database via AssetCenter Web.
- Click the menu item that corresponds to the function in the main menu.

Depending on the selected function, the menu that appears will:







- Present the data from the AssetCenter database.
- Contain a list of choices that will enable you to select data from the AssetCenter database.

- Contain text zones in which you can enter values.
-



The following paragraphs describe what standard functions you can use with AssetCenter Web. The way in which the asset browser function works, which is described later on in this guide, is similar to the other functions.

Available modules

-  General
-  Portfolio
-  Contracts
-  Financials
-  Cable and Circuit
-  Reconciliation

Example of function

To consult the detail of an asset:

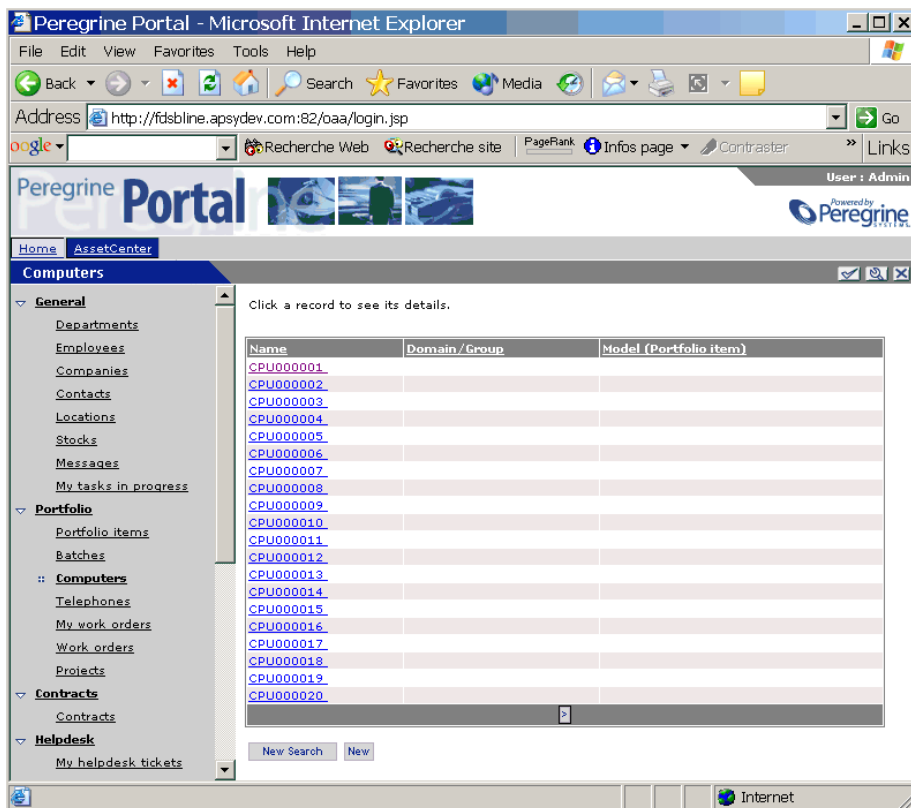
- 1 Click the **AssetCenter** tab and then the **Portfolio items** entry in the main menu of AssetCenter Web.

Figure 8.2. AssetCenter Web - Searching for an asset

- 2 This function enables you to search for an asset depending on certain criteria such as:
 - **Model**
 - **Component of**
 - **Assignment**
 - **User**
 - **Location**
 - **Stock**
 - **Asset tag**
- 3 Click the **Computers** Web application in the **Portfolio** module. Next click **Search**.

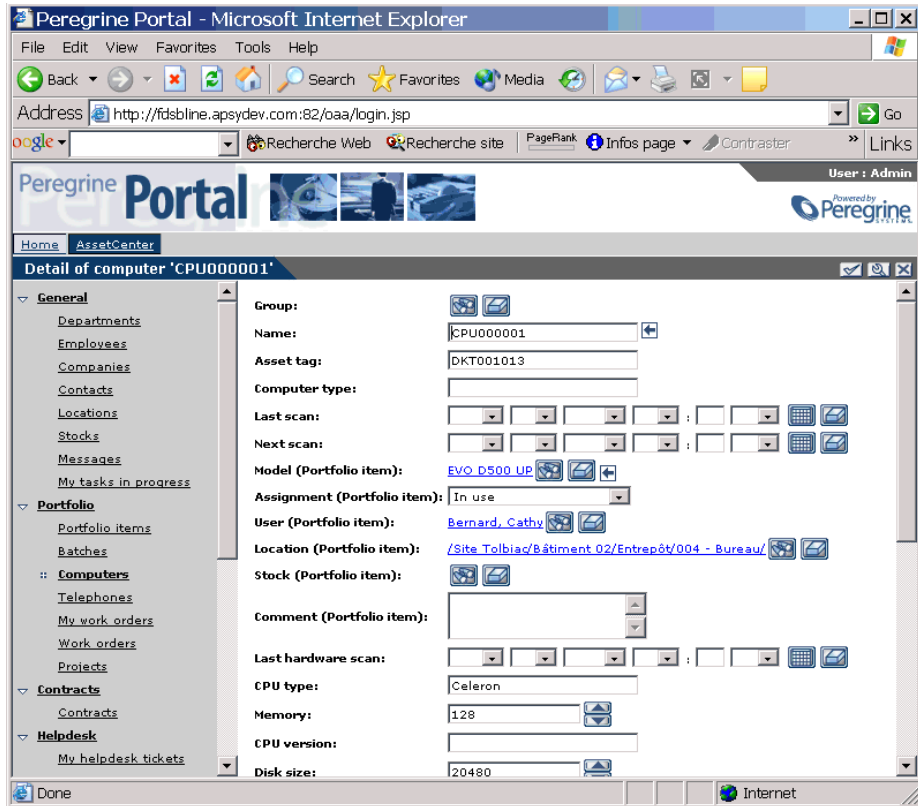
AssetCenter Web displays the list of computers in the portfolio.

Figure 8.3. AssetCenter Web - List of assets



- 4 You can now click on the computer of your choice to display its detail. For example, select **CPU000001** in the **Name** list. The detail of the selected portfolio item is displayed.

Figure 8.4. AssetCenter Web - Detail of a portfolio item



Displayed data

Here is some information on the data processed and displayed by AssetCenter Web.

Live data

The data displayed is identical to the data contained in the AssetCenter database: AssetCenter Web dynamically generates HTML pages so that they contain the most up-to-date data, which reflects the actual situation in the database.

As far as the user is concerned, everything functions exactly as if the AssetCenter database was being accessed via an AssetCenter Windows client.



Note:

After being displayed a certain amount of time, the data might become obsolete due to changes in the database. In order to update it, you need to click the Refresh button of the browser that you are using.


Secure data

The users can only view the data to which they have the right to access. These rights are created and assigned to them by the administrator.

Customizing AssetCenter Web pages

Accessing the customization function for a page

To access the customization of a page:

- 1 Display the page to be customized.
 - 2 Click  on the toolbar.
 - 3 A configuration window is displayed.
-






Note:

Any customizations made in this way are saved at the server level and are valid for all AssetCenter Web users.

Adding a field, link or a section title to an existing page

To add a field, a link or a section title to an existing page:

- 1 Display the page containing the field, link or section title to be customized.
- 2 Click  on the toolbar.
- 3 In the configuration window, select the field, link or "Section Title" in the **Available Fields** list.
- 4 Click  or  between the two lists, or double-click the field, link or section title.



The item is added at the end of the **Current Configuration** list.

You can move the items in the **Current Configuration** list using the  and  icons.

- 5 Click **Save**.

Modifying the attributes of a field, link or section title

To modify the attributes of a field, link or section title:

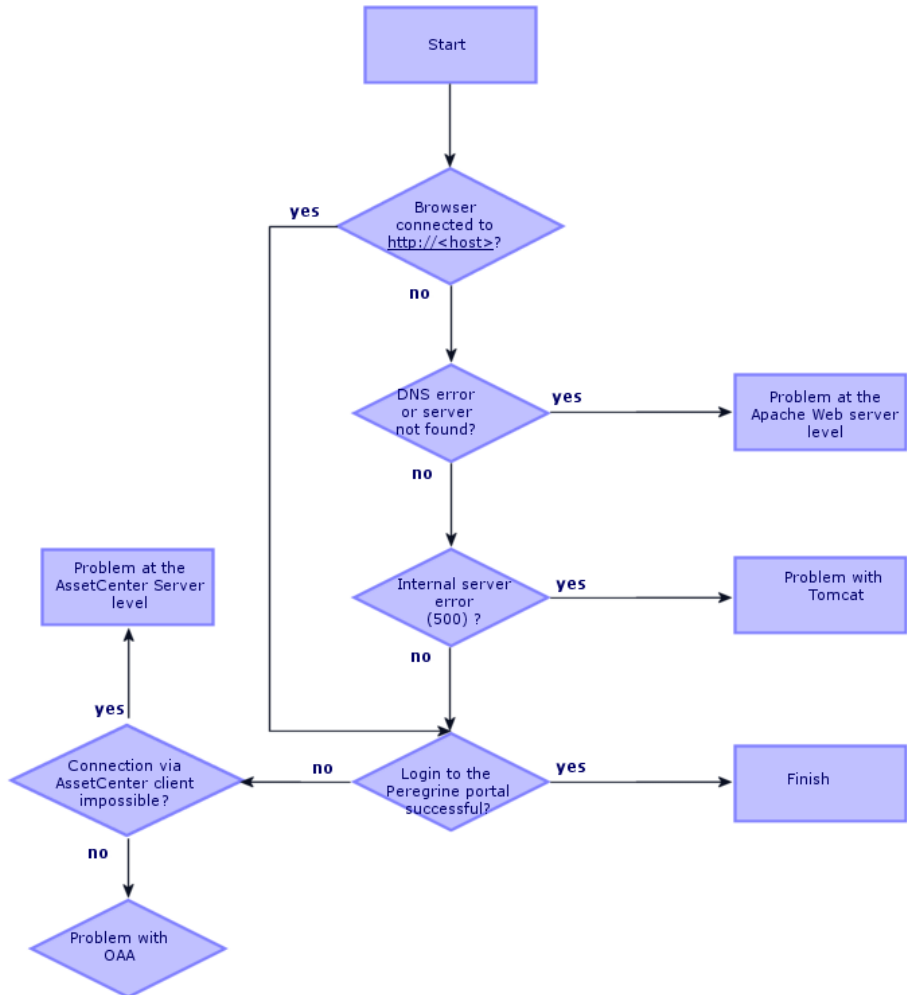
- 1 Display the page containing the field, link or section title to be customized.
- 2 Click  on the toolbar.
- 3 In the configuration window, select the field, link or section title in the **Current Configuration** list.
- 4 Click  to the right of the **Current Configuration** list or double-click the field, link or section title.
- 5 In the second configuration window, modify the attributes of your choice.
- 6 Click **Save**.

9 Troubleshooting

CHAPTER

The following diagram indicates the areas to explore when trying to resolve an installation problem:

Figure 9.1. Troubleshooting - Resolution process



Connecting to the AssetCenter database

If you encounter difficulties connecting to the AssetCenter database from AssetCenter Web (the connection status is Disconnected at the AssetCenter Web administration console level):

- 1 Start the AssetCenter Web administration module with the **System** login.

```
http://<host>/oaa/admin.jsp
```

- 2 Select the **Settings** menu.
- 3 Select the **AssetCenter** tab.
- 4 Make sure the **Database** parameter is correctly set.
- 5 Make sure the login and password you are using to connect in the Web browser match the information in the AssetCenter database.
- 6 Check the ODBC connection to the AssetCenter database.

Depending on the way you run the application server, it looks for either an ODBC User DSN or an ODBC System DSN:

- If you start the application server as a service (the default method), it will reference a System ODBC DSN.
- If you start your application server as an application, it references the User DSN to determine the ODBC connection.

For instructions on how to perform these operations: → [AssetCenter connector \[page 134\]](#).

Apache in Windows

The Apache Web server is not responding

If the Apache Web server is not responding:

- 1 Verify that the network connections are enabled.
- 2 Verify that the **apache.exe** program is running.
- 3 Restart the Apache service.
- 4 Make sure the port that Apache uses is not in use by another network service (Apache uses port 80 by default).

To verify that the network connections are enabled:

- 1 Display the local connection status indicator (**Start/ Settings/ Network Connections/ Local Area Connection** in Windows XP).
- 2 Verify that the status of the connection is **Connected**.

To verify that the apache.exe program is running:

- 1 Press **Ctrl + Alt + Del**.
- 2 Click **Task Manager**.
- 3 In the **Processes** tab, verify that the **apache.exe** program is listed in the **Image Name** column.

To restart the Apache service:

- 1 Start the Windows administration tool (**Start/ Settings/ Control Panel/ Administration Tools/ Services** in Windows XP).
- 2 Locate the **Apache** service in the list and restart it.

To make sure the port that Apache uses is not in use by another network service (Apache uses port 80 by default):

- 1 Stop Apache:
 - 1 Start the Windows administration tool (**Start/ Settings/ Control Panel/ Administration Tools/ Services** in Windows XP).
 - 2 Locate the **Apache** service in the list and stop it.
- 2 Click **Start**.
- 3 Open a DOS command prompt.
- 4 At the command prompt, type `netstart -a` and press Return.
- 5 Make sure the entry with **Proto=TCP,Local Address=<host>:http** does not exist.



This guarantees that when Apache is not running, no service is listening to port http 80.



Note:

Correct the problem, either by modifying Apache's default port, or by disabling/modifying the conflicting service.

Accessing the Web server

If the users cannot access the Web server, even if the server is running and the network and internet connections are enabled:

- 1 Make sure the WINS and DNS servers are installed.
- 2 Look at the error messages in the Apache log file.

To verify that the WINS and DNS servers are installed

To verify that the WINS and DNS servers are installed:

- 1 Start the Windows Add or Remove Programs applet (**Start/ Settings/ Control Panel/ Add or Remove Programs** in Windows XP).
- 2 Click **Add/Remove Windows Components**.
- 3 Select the option **Networking Services**.
- 4 Click the **Details** button.
- 5 Make sure DNS is installed and that the DNS servers are connected and work on the network.

To examine the other error messages in the Apache log files

Display the Apache log files.

By default, the log files are located in the folder **c:\Program Files\Peregrine\Common\Apache2\logs**.

Apache in UNIX

The Apache Web server is not responding

If the Apache Web server is not responding, check the network setup.

To check the network setup:

- 1 Make sure the port that Apache uses is not in use by another network service.



Note:

Apache uses port 80 by default.

You can change this by using the **Port** directive in the **httpd.conf** file.

Use the `netstat` command to list all ports being listened to after shutting down Apache.

```
$ /etc/init.d/oaactl stop
$ netstat -a | grep 80
```

- 2 Make sure the IP address and hostname of the server are configured correctly. If so:
 - The `ping` command successfully gets a response from the server.
 - The `nslookup hostname` displays the correct mapping from the hostname to the IP address.
 - The command `telnet hostname 80` makes a successful connection to the server.

```
$ /usr/sbin/ping hostname -n 5
$ telnet hostname 80
Trying...
Connected to hostname
Escape character is '^]'.
```

Looking at the Apache log files for advanced errors

If you are having difficulties with the Apache Web server, take a look at the log files.

To examine the Apache log files for advanced errors:

Open the Apache log files.

By default, the Apache log files are located in the **<Base installation folder>/peregrine/common/apache2/logs** folder.

IBM HTTP Server

The AssetCenter Web installer creates duplicate alias entries in the IBM HTTP Server when you install more than one Peregrine OAA Platform application on WebSphere.

Duplicate entries can also occur if you reinstall AssetCenter Web or install another Peregrine OAA Platform application on a system that formerly had AssetCenter Web installed on it.

Delete the duplicated aliases from the **httpd.conf** file of IBM HTTP Server.

Tomcat

Before you can troubleshoot problems on Tomcat, you must become familiar with starting and stopping Tomcat on your operating system.

You also need to know the location of the Tomcat log files.

Stopping and starting Tomcat in Windows

To stop or start Tomcat in Windows:

- 1 Start the Windows administration tool (**Start/ Settings/ Control Panel/ Administration Tools/ Services** in Windows XP).
- 2 Locate the **Peregrine Tomcat** service in the list.
- 3 Use the buttons to stop, start or restart it.

Stopping and starting Tomcat in UNIX

```
/etc/init.d/oaactl<start/stop/restart>
```

The following table lists the default locations of the Tomcat log files:

Table 9.1. Tomcat in UNIX - Location of log files

Operating system	Default location of the Tomcat log files
Windows	C:\Program Files\Peregrine\Common\Tomcat4\logs
UNIX	/ root installation folder /peregrine/common/tomcat4/logs

Checking for Tomcat port conflicts

The following table displays the default Tomcat port usage:

Table 9.2. Tomcat in UNIX - Default ports

Port number	Tomcat service
8005	Tomcat Administration
8009	Tomcat AJP13 Worker Port

In the Tomcat log file, **stderr.log**, the following line indicates the AJP13 port currently being used:

```
[INFO] ChannelSocket - -JK2: ajp13 listening on tcp port 8009
```

To check for Tomcat port conflicts:

- 1 Stop the Tomcat service.
- 2 Use `netstat -a` to list ports being listened on. Check for port conflicts.
- 3 Make necessary modifications to Tomcat port configuration or disable (or modify) the conflicting service.

Additional information about Tomcat is available at:

```
http://jakarta.apache.org/tomcat/
```

Checking for port conflicts: an example

Check for entries where the value of **Proto** is **TCP** and the **State** and **Listening**. For example, the following result for the `netstat -a` command shows that the ports 80, 8009, 8025, 12670 and 1585 are being used:

```

Active Connections
Proto Local Address Foreign Address State
TCP 0.0.0.0:8 0 0.0.0.0:0 LISTENING
TCP 0.0.0.0:8009 0.0.0.0:0 LISTENING
TCP 0.0.0.0:8025 0.0.0.0:0 LISTENING
TCP 0.0.0.0:12670 0.0.0.0:0 LISTENING
TCP1 0.2.3.154:1032 66.163.173.77:80 ESTABLISHED
TCP1 0.2.3.154:1342 10.2.3.154:12670 ESTABLISHED
TCP1 0.2.3.154:1585 0.0.0.0:0 LISTENING
TCP1 0.2.3.154:1585 10.2.0.112:139 ESTABLISHED

```

Check for Tomcat errors

Make sure that you are working with clean files.

To ensure a clean environment for troubleshooting:

- 1 Shutdown the Apache and Tomcat services.
- 2 Remove all log files.
- 3 Restart the Apache and Tomcat services.
- 4 Use a browser to connect to the Web server.

mod_jk.log file

This file contains log information regarding the out-of-process TCP connection between the Apache Web server and Tomcat.

This file is empty when there are no errors. It contains hints about connection failures when the AJP13 port is in conflict with another service, or when the **Tomcat mod_jk** connector is configured incorrectly.

stdout.log file

The following extract of the **stdout.log** file is a normal output.

```

Bootstrap: Create Catalina server
Bootstrap: Starting service
Starting service Tomcat-Standalone
Apache Tomcat/4.1.12
Instantiating Archway Servlet...
2002-12-10 12:22:13,079 INFO [main] - Using application preferences in
/C:/Program Files/Peregrine/Common/Tomcat4/webapps/oa/WEB-INF/local.xml
2002-12-10 12:22:13,119 INFO [main] - Using default preferences in /C:/
Program
Files/Peregrine/Common/Tomcat4/webapps/oa/WEB-INF/default/archway.xml
2002-12-10 12:22:13,200 INFO [main] - Using default preferences in /C:/
Program

```

```
Files/Peregrine/Common/Tomcat4/webapps/oa/WEB-INF/default/common.xml
2002-12-10 12:22:13,240 INFO [main] - Using default preferences in /C:
/Program
Files/Peregrine/Common/Tomcat4/webapps/oa/WEB-INF/default/logging.xml
2002-12-10 12:22:13,270 INFO [main] - Using default preferences in /C:
/Program
Files/Peregrine/Common/Tomcat4/webapps/oa/WEB-INF/default/themes.xml
2002-12-10 12:22:13,280 INFO [main] - Using default preferences in /C:
/Program
Files/Peregrine/Common/Tomcat4/webapps/oa/WEB-INF/default/xsl.xml
Bootstrap: Service started
```

Look for the following in this file during an error:

- Archway Servlet is not instantiated.
- The **webapps** location is incorrect.
- **Bootstrap** service failed to start.

stderr.log file

The following extract of the **stderr.log** file is a normal output.

```
Created catalinaLoader in: C:\Program Files\Peregrine\Common\Tomcat4\s
erver\lib
[INFO] Registry - -Loading registry information
[INFO] Registry - -Creating new Registry instance
[INFO] Registry - -Creating MBeanServer
[INFO] ChannelSocket - -JK2: ajp13 listening on tcp port 8009
[INFO] JkMain - -Jk running ID=0 time=0/120 config=C:\Program
Files\Peregrine\Common\Tomcat4\conf\jk2.properties
```

Look for the following in this file during an error:

- **catalinaLoader** was not created or is pointing to an incorrect location.
- **ChannelSocket - JK2: ajp13** failed to connect or is connecting on an incorrect port number.
- **JkMain** is not using the right **jk2.properties** file.

localhost_log.<date>.txt file

There should not be any Java errors in this log file.

This file logs application manager activity in deploying Peregrine OAA Web applications.

OAA

If you are having trouble with your Peregrine OAAWeb application, verify your the AssetCenter database server and view the OAA logs.

AssetCenter database server configuration

Make sure that AssetCenter Web is connecting to the AssetCenter database server and that it is currently functional.

To check the AssetCenter database server configuration:

- 1 Connect to the AssetCenter Web administration console (**admin.jsp**).
- 2 Open a session with the **System** login. A password is not required unless you have specified one since installation.
- 3 From the administration console, check the connection status of the listed connectors.
- 4 Click on the target for the database server (**ac**).
- 5 Verify that the host and port for the database server are correct.

OAA log files

The following table lists the default locations of the OAA log files.

Table 9.3. OAA - Log files

Operating system	Default location of the Tomcat log files
Windows	C:\Program Files\Peregrine\Common\Tomcat4\bin\arch-way.log
UNIX	/<root installation folder>/peregrine/common/tomcat4/arch-way.log

Make sure the log file contains the following elements:

- A listing of installed OAA components and their version numbers.
- A correct listing of registered packages.
- An **Archway initialization complete** statement.

If the file contains **Java ClassNotFound** exceptions, check to see if all the required **.jar** files are found.

WebSphere

Duplicated aliases

Aliases are sometimes duplicated in the **httpd.conf** file of IBM HTTP Server when installing WebSphere.

If this happens, the administration page **http://<Web server name>:<Port>/oaa/admin.jsp** does not display correctly.

In the **httpd.conf** file, remove any duplicate lines starting with:

```
Alias /oaa
```

By default, the **httpd.conf** file is located in the **conf** folder of the IBM HTTP Apache Web server.

There must only remain one instance of the following lines:

```
Alias /oaa "C:\WebSphere\AppServer\installedApps\oaa.ear\portal.war"  
Alias /oaa "C:/WebSphere/AppServer/installedApps/oaa.ear/portal.war"
```

Index

Index

A

- ac (connector), 140 , 133
- Access rights, 138
- acw.* (user rights), 137
- Additional documentation, 12
- admin.jsp, 131
- Administration (menu), 132
- Administration (tab), 26
- Administration - portal, 131
- amdb.ini, 134
- Apache, 18
 - Replacement by WebSphere 4.0
 - UNIX, 72
 - Windows, 39
 - Replacement by WebSphere 5.0
 - UNIX, 79
 - Windows, 47
 - Replacement with WebLogic
 - UNIX, 83
 - Windows, 52
 - Server not responding
 - UNIX, 160
 - Windows, 157
 - Starting, 130
 - Version already installed
 - UNIX, 70
 - Windows, 30
- apache.exe, 158 , 158
- apachemonitor.exe, 130

Application server

- Alternative to Tomcat
 - UNIX, 69
 - Windows, 29
 - Load balancing (See Load balancing)
- archway.log, 127 , 115
- AssetCenter, 18
 - Demonstration database, 143
- AssetCenter (connector), 134
- AssetCenter (tab), 147 , 145 , 136 , 134 , 134
- AssetCenter API, 20
- Authentication - migration, 27
- autorun.exe, 62 , 60

B

- bin (folder), 129
- Browsing - window, 145

C

- Cable and Circuit (module), 147
- Calendars, 146
- Client - database access, 18
- Common (tab), 140 , 135 , 51 , 26
- Components
 - Custom installation
 - UNIX, 91
 - Windows, 61
 - Typical installation
 - UNIX, 87

- Windows, 58
- Components of AssetCenter Web, 18
- Configuration, 131-141
 - Default values, 140
 - Modification, 140
- Connected (status), 140
- Connecting to AssetCenter Web, 144
- Context root, 99
- Contracts (module), 147
- Control panel (menu), 140
- Control Panel (module), 132
- Custom (See Installation)
- Custom installation
 - Procedure
 - UNIX, 92
- Customization
 - Icon, 146
 - Migration, 25
 - Pages, 152
 - Access, 152
 - Adding, 153
 - Modification, 153

D

- Database
 - Access, 18
 - Connection - problems, 157
 - Location, 20
- Database (setting), 134 , 134
- Data - updating, 152
- Dates, 146
- Default Capability Words (setting), 136
- Demonstration database, 143
- Disconnected (status), 133
- Disconnection, 146
- DLL - UNIX environment variables, 98
- Documentation
 - Web site, 15
- Duplicated aliases, 166

E

- Enable user registration (setting), 26
- Encoding (section), 135

F

- Fields - deleting the contents, 146
- Financials (module), 147
- Functionality, 143

G

- General (module), 147

H

- Heap size for WebSphere
 - UNIX, 76
 - Windows, 44
- Home (tab), 145
- httpd.conf, 161 , 110 , 79 , 78 , 72 , 71 , 49 , 47 , 39 , 31
 - Duplicated aliases, 166

I

- IBM HTTP Server alias, 79 , 72 , 47 , 39
- IBM HTTP Server - problems, 161
- Icons
 - Pages, 146
 - Toolbar, 145
- IIS
 - With Tomcat
 - Windows, 32
 - With WebLogic
 - UNIX, 84
 - Windows, 52
- iisforward.dll, 55
- iisproxy.dll, 56
- Installation
 - Custom
 - Choice, 21
 - Development environment, 23
 - Production environment, 24
 - UNIX, 91
 - Windows, 61
 - Strategies, 21-28
 - Tests
 - UNIX, 101
 - Windows, 66
 - Typical

- Choice, 21
- Development environment, 22
- Procedure, 88 , 60
- Production environment, 24
- Scripts, 88
- UNIX, 86
- Windows, 58
- UNIX, 69-101
- Windows, 29-67
- installservice.bat, 125 , 113
- Interface, 143
 - Example, 147
- ISAPI
 - Filter, 33
 - IIS, 33
 - Windows, 55
- isapi_redirector2.dll, 119 , 33

J

- jaas.jar, 55
- jai_codec.jar, 55
- jai_core.jar, 55
- jakarta (virtuel directory), 118 , 34
- jar (file), 66 , 55
- Java, 18
- JAVA_HOME (environment variable), 61
- jce1_2_1.jar, 55
- jcrt.jar, 55
- jk2.properties, 164 , 124 , 113
- jk2.reg, 118 , 33
- jnet.jar, 55
- js.jar, 80 , 49
- jsp (files) - migration, 26
- jsse.jar, 55

L

- Languages, 135
- libaamapi43.so, 134
- Lists - searches, 146
- Load balancing, 103-127
 - Principles, 103
 - Tomcat for Apache, 105
 - Tomcat for IIS, 115

- local_policy.jar, 55
- localhost_log*.txt, 164
- log4j-1.2.6.jar, 55
- login.jsp, 71 , 31
- Login (page) - migration, 27
- Login page - Display error, 72

M

- mail.jar, 80 , 49
- Main menu, 145
- Memory (option), 65
- Migration, 25
 - Authentication methods, 27
 - Customized .jsp files, 26
 - Login pages, 27
 - Personalized pages, 26
 - Processes, 28 , 25
 - Skins, 26
 - Stylesheets, 26
 - Tailoring Kit, 27 , 27
 - Themes, 26
- mllibwrapper_jai.jar, 55
- mod_jk.conf, 109 , 70 , 36 , 30
 - Supported versions, 71 , 31
- mod_jk.dll, 70 , 30
- mod_jk.log, 163
- Modules, 147

O

- OAA, 18
 - Configuration, 131
 - Log files, 165
 - Problems, 165
 - Updating, 88
- oaa (virtual directory), 120 , 35
- oaactl, 90
- oaasecurityproxy.jar, 55
- Obsolete data, 152
- Overview, 17

P

- Pages
 - (See Also Customization)

- Parameters
 - Validation, 140
 - Password, 26
 - Personalization
 - Pages
 - Migration, 26
 - Rights, 139
 - Personalization (option), 140
 - Poller (See Script pollers)
 - pop3.jar, 80 , 49
 - Portfolio (module), 147
 - Portal (tab), 28
 - portal*.war, 96 , 73 , 48 , 41
 - Ports
 - Conflicts
 - UNIX, 91
 - Windows, 62 , 38
 - Custom installation
 - UNIX, 91
 - Windows, 61
 - Typical installation
 - UNIX, 87
 - Windows, 59
 - Problems
 - Documentation Web site, 15
 - Resolution, 155-166
 - Technical support, 14
 - profile.scr, 137
 - Profile (tab), 138 , 137
- R**
- Reconciliation (module), 147
 - Refreshing data, 152
 - Reset Server (button), 141 , 140
- S**
- Script pollers
 - Tomcat
 - Disabling, 106
 - Enabling, 114
 - SDK Java2 Standard Edition, 18
 - Searches, 146
 - Security, 152
 - server.policy, 54
 - server.xml, 122 , 110 , 37 , 36
 - Services, 59
 - Setting CATALINA_OPTS (window), 64
 - startupserver.sh, 100
 - startWebLogic.cmd, 84
 - startweblogic.cmd, 53
 - stderr.log, 164 , 162
 - stdout.log, 163
 - sunjce_provider.jar, 55
 - Supported environments, 19
- T**
- Tailoring Kit - migration, 27 , 27
 - Technical support, 14
 - Tomcat, 18
 - Ports - conflicts
 - UNIX, 162
 - Replacement by WebSphere 4.0
 - UNIX, 72
 - Windows, 39
 - Replacement by WebSphere 5.0
 - UNIX, 79
 - Windows, 47
 - Replacement with WebLogic
 - UNIX, 83
 - Windows, 52
 - Starting, 129
 - UNIX, 161
 - Windows, 161
 - Stopping
 - UNIX, 161
 - Windows, 161
 - Update, 88
 - Version already installed
 - UNIX, 70
 - Windows, 30
 - With IIS, 32
 - Tomcat memory settings (window), 65
 - Toolbar, 145
 - Training, 15
 - Translation
 - Icon, 146

- UNIX, 83
- Windows, 52
- Typical (See Installation)

U

- Uninstallation
 - UNIX, 100
 - Windows, 65
- UNIX environment variables, 98
- Update, 60
- Updating
 - Data, 152
 - OAA, 88
- URL, 144 , 131
- us_export_policy.jar, 55
- User profiles (menu), 137
- User registration, 26
- User rights, 136
- User rights (menu), 137
- User rights (tab), 137
- Using, 143-153

V

- Virtual directory
 - jakarta, 34
 - oaa, 35
 - WebLogic
 - UNIX, 85
 - Windows, 57
 - WebSphere
 - UNIX, 77
 - Windows, 46

W

- weblication (connector), 140 , 133
- WebLogic
 - UNIX, 83
 - Windows, 52
- Web server
 - Access - Problems, 159
 - Alternative to Apache
 - UNIX, 69
 - Windows, 29

- Implementation, 129
- WebServer plug-in
 - Regeneration, 78 , 46
- WebSphere
 - Configuration
 - Linux, 99
 - Solaris, 99
 - Problems, 166
- WebSphere 4.0
 - Configuration
 - UNIX, 72
 - Windows, 39
 - Installation, 40
 - UNIX, 72
 - UNIX installation, 73
 - Windows, 39
- WebSphere 5.0
 - Configuration
 - UNIX, 79
 - Windows, 48
 - Installation, 47
 - UNIX, 79
 - Windows, 47
- WebSphere Translation Server
 - UNIX, 81
 - Windows, 50
- workers.properties, 107 , 70 , 30
- workers2.properties, 36
- Work session - ending, 145
- wts.jar, 82 , 51

