Oracle® WebCenter Interaction

Maintenance Guide

10g Release 3 (10.3)

November 2008

ORACLE

Oracle WebCenter Interaction Maintenance Guide, 10g Release 3 (10.3)

Contents

A. Java Virtual Machine Configuration

CHAPTER 1

Welcome

This book describes strategies for maintaining an Oracle WebCenter deployment.

For an overview of all deployment documentation, see the *Oracle WebCenter Interaction Deployment Overview Guide*. For products and versions covered by this book, see the section in that guide titled "Products Covered by the Deployment Guide."

How to Use This Book

Audience

This guide is written to provide guidance to people responsible for maintaining the Oracle WebCenter system. Access to resources with strong knowledge of the platform operating system, database, web and application servers, and any other third-party software is recommended.

Organization

This guide includes the following chapters:

- This chapter provides information on how to use this guide and describes general resources available to assist in the Oracle WebCenter deployment.
- Chapter 2, "Developing a Production Maintenance Plan" provides an overview of portal maintenance tasks and tools.

- Chapter 3, "Performance Tuning" details the process of tuning application servers and standalone Oracle WebCenter components to the needs of your Oracle WebCenter deployment.
- Appendix A, "Java Virtual Machine Configuration" describes how to adjust JVM memory parameters and turn garbage collection logging on and off.

Typographical Conventions

This book uses the following typographical conventions.

Table 1-1 Typographical Conventions

Convention	Typeface	Examples/Notes
• Items you need to take action on (such as files or screen elements)	**bold**	• Upload **Procedures.doc** to the portal. • To save your changes, click **Apply Changes**.
• User-defined variables • New terms • Emphasis • Object example names	*italic*	• The migration package file is located in *install_dir*\serverpackages. • *Portlets* are Web tools embedded in your portal. • The URI *must* be a unique number. • The example Knowledge Directory displayed in Figure 5 shows the *Human Resources* folder.

• Text you enter
• Computer generated text (such as error messages)
• Code samples
	`computer`	• Type `Marketing` as the name of your community.
		• This script may generate the following error: `ORA-00942 table or view does not exist`
		• Example:
		`<setting name="SSOCookieIsSecure">`
		`<value xsi:type="xsd:integer">0</value>`
		`</setting>`

• Environment variables	ALL_CAPS	• ORACLE_HOME specifies the directory where Oracle products are installed.

Oracle Documentation and Resources

This section describes other documentation and resources provided by Oracle.

Table 1-2 Oracle Documentation and Resources

Resource	Description
Installation and Upgrade Guides	These guides describe the prerequisites (such as required software) and procedures for installing or upgrading the various Oracle WebCenter components. These guides are available on the Oracle Technology Network at http://www.oracle.com/technology/documentation/bea.html.
Release Notes	The release notes provide information about new features, issues addressed, and known issues in the release of various Oracle WebCenter products. They are available on the Oracle Technology Network at http://www.oracle.com/technology/documentation/bea.html.
Administrator Guides	These guides describe how to manage, maintain, and troubleshoot the various Oracle WebCenter components. These guides are available on the Oracle Technology Network at http://www.oracle.com/technology/documentation/bea.html.
Online Help	The online help is written for all levels of Oracle WebCenter users. It describes the user interface for Oracle WebCenter components and gives detailed instructions for completing tasks in Oracle WebCenter products. To access online help, click the help icon.
Oracle Technology Network (OTN)	The Oracle Technology Network is the world's largest community of developers, DBAs, and architects using Oracle products and industry-standard technologies. Every day, members collaborate via OTN to share real-world insight, expertise, and best practices on how to build, deploy, manage, and optimize applications. As a member of the Oracle Technology Network you will enjoy access to software downloads, discussion forums, documentation, wikis, podcasts, blogs, plus much more. Access the Oracle Technology Network at http://www.oracle.com/technology/index.html.
Oracle Support	The Oracle Support site provides access to all Oracle support resources including online support, software and patches, technical articles, and contact numbers. Access the Oracle Support site at http://www.oracle.com/support/index.html.

CHAPTER 2

Developing a Production Maintenance Plan

This chapter provides an overview of Oracle WebCenter maintenance tasks and tools.

The purpose of this chapter is to help you scope administrative responsibilities for the Oracle WebCenter so that you can develop a maintenance plan.

This chapter includes the following topics:

Periodic Tasks

The following table provides suggestions for periodic tasks that you should consider as part of your production system maintenance plan.

Frequency	Task
Daily	Modify security of portlets, communities, and other objects in the portal. Modify permission roles for users. Publish new and existing applications/portlets to remote servers. Monitor portal, database, and remote servers alerts for CPU, memory, and hard disk usage to ensure availability.
Weekly	Install releases to one or more software components.
Monthly	Add new hardware to the environment (for example, new remote servers, new hard disk, and so on).
Ad Hoc	Install Oracle WebCenter patches. Install server patches from critical third-party software providers, such as operating system and anti-virus software.

Monitoring Oracle WebCenter Services

The Counter Monitoring System collects information from various performance counters for portal applications and exposes them for diagnosis and review. This system can be used to examine counters from any Oracle WebCenter application that resides on a remote host, provided the both the remote host and the counter monitoring system are on a network in which they can reach each other via UDP.

With the Counter Monitoring System you can:

- Set up counter logging files in your desired format to view counter information.
- Use the Counter Monitoring console to request specific counter data in real time.
- Use the Windows Perfmon utility to view portal counter data, if you use a Windows system.

For detailed information on the Counter Monitoring System, see the *Administrator Guide for Oracle WebCenter Interaction*.

Monitoring Databases and Java Application Servers

Databases support Performance Monitor counters on Windows. WebLogic, Tomcat, and WebSphere do not. For information on performance monitoring for application servers, refer to the related application server documentation.

Monitoring Usage

Oracle WebCenter Analytics is an advanced usage tracking and analytics tool designed exclusively for Oracle WebCenter. This portal add-on enables you to assess portal ROI and define future opportunities with usage trends in mind. Oracle WebCenter Analytics delivers the following features out of the box:

- Usage Tracking Metrics: Tracks metrics for common portal functions, including community, portlet, collaboration project, and document hits, as well as search queries, logins, and more.
- Behavior tracking: Tracks usage patterns, such as number and duration of visits.
- User Profile Correlation: Correlates metrics with user profile information. In this way, usage tracking reports can be viewed and filtered by profile data, such as country, company and department.

Oracle WebCenter Analytics includes the following reports that you can customize by setting filtering, grouping, and presentation options.

Report	Description	Features
Community Traffic	Displays traffic information for each community in the portal.	Displays traffic in three ways: • Hits: Count of page views within the community. • Visits: Count of visits to the community, each visit can consist of several hits. • Users: Count of unique users who have visited the community. Users can select to see the most active, least active, or a select list of communities.
Community Response Time	Displays average, maximum and minimum response time for each community within the portal.	Calculates response time as the time between the portal receiving a community page request until the time an HTML response is sent to the client. Users can select to see the slowest response times, fastest response times, or response times for a select list of communities.

Report	Description	Features
Portlet Usage	Shows usage statistics within gatewayed portlets.	Displays traffic in two ways: • Activity: Count of hits on an object (for example, a button or link) within a portlet. • Users: Count of unique users who have performed an activity within the portlet. Users can select to see the most active, least active, or a select list of portlets.
Portal Traffic	Shows an aggregate of all portal page views within the portal.	
Portal Users	Displays statistics regarding portal user accounts.	Displays the following four figures to help explain user inception and activity. • Total user accounts in the portal. • Added (new) user accounts created in the portal during a given date range. • Active users defined by activity during a given date range. • Inactive users defined by inactivity during a given date range
Portal Logins	Shows an aggregate of all portal logins.	
Portal Duration	Displays the length of visits to the portal.	Calculate visit durations as the time between login and logoff or the time between login and inactivity for a configurable length of time. This report shows both average and maximum visit duration.

Report	Description	Features
Search Keywords	Shows the top search keywords entered in searches within the portal.	See the top 5, 10, 25, 50 or 100 search keyword phrases entered within the portal.
Document Views	Shows statistics for document views in the portal.	Can display these statistics in two ways: • Top Documents: List of top documents viewed with view count.· • Folders: Count of all document views by folders in the knowledge directory.

Troubleshooting Tools

This section describes logging and troubleshooting tools. It includes the following topics:

Oracle WebCenter Logging Utilities

Oracle WebCenter Logging Utilities includes three *log message receivers* that allow for a wide variety of logging solutions. In the OpenLog Framework, log message receivers act to display or store log messages generated by *log message senders*, such as the portal, Oracle WebCenter Collaboration, or Oracle-BEA AquaLogic Publisher. Oracle WebCenter Logging Utilities include:

- Logging Spy. Previously called PTSpy, this utility is the primary log message receiver for the OpenLog Framework. In addition to displaying log messages from the portal and other Oracle WebCenter products and services, Logging Spy provides fine-grained filtering, viewing of saved log files, highlighting of errors, and the searching and sorting of log messages.
- Logger. Logger runs as an unattended background process that receives log messages from the OpenLog Framework and writes the messages to the file system. In addition to this primary use, the Logger can be configured to provide output in other ways, such as sending log messages to an e-mail system.

- Console Logger. The Console Logger runs in a console window, writing log messages to the console standard output. The Console Logger has limited use; in most cases, it is preferable to use Logging Spy.

For information on using these utilities, see the *Installation Guide for Oracle WebCenter Logging Utilities*.

View Source

HTML code creates Web pages. In turn, Oracle WebCenter Activity Spaces generate HTML code. Along with HTML from the View and Display pages, the underlying framework inserts some general information for each page. If there is an error on the page, the Error framework might insert additional debugging information. You can review the HTML source for any given Web page to gather this information. Often the HTML for a given error page contains detailed information about the error.

When to Use View Source

Use View Source to gather more information when you receive an error on a portal page or when you want some general information about the page. For example, use View Source if you receive the following error message on a portal page: “An unexpected error occurred when trying to start the Editor.” The message itself gives no clues to the source of the error, but when you view the HTML source code for the page, you might be able to determine the source of the error.

How to Use View Source

While viewing the Web page, in the browser menu, click **View** | **Source**. This displays the HTML for that page. If the browser menu is unavailable, sometimes it is possible to view source by right-clicking the page and then clicking **View Source**. With this approach, be aware that if there are frames, only the source for the frame in which you right-clicked will display. When the source displays, you can search for specific pieces of information as described in the next section.

What Is Available in View Source

Each portal page contains several pieces of general information:

- To determine the server hosting the portal, search for “Hostname:”. The hostname of the server is commented in the source: “<!--Hostname: My Server-->”.
- To find information about the build of the portal, search for “Portal Version:”, “Clingiest:”, and “Build Date:”.

- To find information about general timing data points, search for “Total Request Time:”, “Control Time:”, “Page Construction Time:”, and “Page Display Time:”.

If there is an error on the page, View Source might provide extended information. There are three items that you can search for:

- To view the error, search for “alert Error Title”. You might have to repeat the search because several error related Tanglements might use that text.
- To view extended information, search for “Extended Error Message:”. The extended error is wrapped in an HTMLComment and thus does not show up on the page, “<!--Extended Error Message: Sample Extended Error message.-->”. The extended information, controlled by the developer and Activity Space, is frequently the same as the error message that displays in the user interface.
- You might also need to search for “unexpected error”. When the portal encounters an unexpected error, the stack trace for the error is often inserted into an HTMLComment. The following example informs the user where the error originates from. The user then has a starting point from which to perform further debugging:

```
<!--An unexpected error occurred when trying to start the Editor.:

com.plumtree.openfoundation.util.XPException: An unexpected error
occurred when trying to start the Editor.

at
com.plumtree.portalpages.admin.editors.group.GroupModel.DoTaskOnStartEd
itor(GroupModel.java:411)
```

CHAPTER 3

Performance Tuning

This chapter details the process of tuning application servers and standalone Oracle WebCenter products to the needs of your Oracle WebCenter deployment.

The standalone Oracle WebCenter products are:

- Oracle WebCenter Analytics
- Oracle WebCenter Interaction Automation Server
- Oracle WebCenter Collaboration
- Oracle WebCenter Interaction Document Repository
- Oracle WebCenter Interaction Notification
- Oracle WebCenter Interaction PTUpload
- Oracle-BEA AquaLogic Interaction Publisher
- Oracle-BEA AquaLogic Interaction Studio
- Oracle-BEA AquaLogic Interaction Workflow

Tuning a Java Application Server or Standalone Oracle WebCenter Product

For Java application servers and standalone Oracle WebCenter products, tuning is a matter of adjusting various Java Virtual Machine (JVM) settings to optimize garbage collection. Sun provides a comprehensive document on this subject, *Tuning Garbage Collection with the 1.4.2 Java[tm] Virtual Machine*, which you can find at http://java.sun.com/docs/hotspot/gc1.4.2/**.**

The following provides a brief background on the garbage collection process and a detailed, Oracle WebCenter focused process for tuning JVM garbage collection.

Garbage Collection Concepts

Garbage collection is the process the JVM undergoes to remove unused objects from memory. The following description of the garbage collection process is simplified for the purpose of this guide.

The JVM stores objects in two sections of the heap: the *young generation* and the *tenured generation***.** The young generation is where objects are first created and provides the quickest, least CPU intensive access to objects. When the young generation fills, older active objects are transfered to the larger tenured generation. Objects in the tenured generation are more CPU intensive to access than those in the young generation.

The JVM undertakes two types of garbage collection. The *minor collection* runs when the young generation fills. It clears garbage objects and copies surviving objects to the tenured generation. The *major collection* runs when the tenured generation fills. The minor collection is significantly less CPU-intensive than the major collection.

Garbage Collection Logs

In order to analyze garbage collection impact on application server performance, a garbage collection log needs to be collected. The process is:

1. Enable garbage collection logging in the JVM. This is done in different places for each of the supported application servers and standalone Oracle WebCenter products. For details on enabling garbage collection logging, see Appendix A, “Java Virtual Machine Configuration.”
2. Restart the application service to start logging garbage collection memory usage.
3. Run the until the problem occurs. If the problem is continuous, collect approximately a 24 hours of data.

Note: Every time the application server is restarted, the garbage collection log is overwritten. It is important to turn off automatic restarting of services, especially if you are investigating an issue that yields a server crash.

Analyzing the Garbage Collection Log

Tagtraum industries (http://tagtraum.com) provides a free utility, *gcviewer*, for analyzing garbage collection logs generated by the JVM. Load the garbage collection log into gcviewer and determine which issue is occurring based on the descriptions in Table 3-1.

Table 3-1 Garbage Collection Performance Issues

Issue	Symptoms in Garbage Collection Log	Impact of the Issue
Insufficient total (heap) memory allocated	Memory usage trends upwards and reaches the top of the total memory allocated.	Reduces the performance or potentially crashes the Oracle WebCenter product.
Excessive total (heap) memory allocated	Memory usage peaks much lower than total memory allocated.	Can cause a slowdown across all applications on the server. The application server or Oracle WebCenter product is taking up too much of the system memory.
Insufficient young generation memory allocated	Sawtoothed memory usage.	Reduces the performance of the Oracle WebCenter product. This represents excessive minor garbage collector runs, which increases the number of objects in the tenured generation. Objects in the tenured generation are more resource intensive when called.

Resolving Garbage Collection Performance Issues

Resolving the issues described in Analyzing the Garbage Collection Log is a matter of adjusting the JVM memory settings and reanalyzing the garbage collection log. Table 3-2 shows what memory settings to adjust for each issue. For details on how to adjust these settings for each supported application server and standalone Oracle WebCenter product, see Appendix A, "Java Virtual Machine Configuration."

Table 3-2 Garbage Collection Performance Issue Resolution

Issue	Resolution	JVM Memory Parameter
Insufficient total (heap) memory allocated	Increase total heap memory allocation until memory usage stays reasonably below total memory.	Increase `-Xmx`
Excessive total (heap) memory allocated	Decrease total heap memory allocation until memory usage is reasonably, but not excessively, below total memory.	Decrease `-Xmx`
Insufficient young generation memory allocated	Increase young generation memory allocation until the memory usage trend is horizontal.	Adjust `-XX:NewRatio`

APPENDIX A

Java Virtual Machine Configuration

This appendix describes how to adjust JVM memory parameters and turn garbage collection logging on and off. Instructions below cover applications supported by Oracle WebCenter and those standalone Oracle WebCenter products.

The standalone Oracle WebCenter products are:

- Oracle WebCenter Analytics
- Automation Server
- Oracle WebCenter Collaboration
- Document Repository
- Notification
- PTUpload
- Oracle-BEA AquaLogic Interaction Publisher
- Oracle-BEA AquaLogic Interaction Studio
- Oracle-BEA AquaLogic Interaction Workflow

Java Memory Switches

The following are Java memory switches used to tune JVM garbage collection. Use these switches in conjunction with the instructions specific to your application server or Oracle WebCenter product.

- `-Xloggc:<path/filename>`

  This switch turns on garbage collection logging for the JVM. Replace **`<path/filename>`** with the location where the garbage collection log should be generated.

- `-Xms` and `-Xmx`

  These switches set the minimum (`-Xms`) and maximum (`-Xmx`) heap size for the JVM. The JVM adjusts heap size based on object usage and bounded by these two switches. Setting these switches to the same value increases predictability by removing the ability of the JVM to adjust the heap size.

  Caution: Fixing the heap size to a specific value requires special attention to memory tuning.

- `-XX:NewRatio`

  This switch sets the ration of the young generation to the tenured generation. For example

  `-XX:NewRatio=3`

  would mean that the tenured generation is 3x the size of the young generation, or, in other words, the young generation is one quarter of the heap and the tenured generation is three-quarters of the heap.

Application Servers

Tomcat 5.x

To update Java options for Tomcat 5.x on Windows:

1. Run **TOMCAT_HOME\tomcat5w.exe**
2. Click the **Java** tab.
3. Update the Java memory switches in the **Java Options:** box.
4. Click **OK**. Restart the Tomcat service.

Tomcat 6.x

To update Java options for Tomcat 6.x on Windows:

1. Run **TOMCAT_HOME\tomcat6w.exe**
2. Click the **Java** tab.
3. Update the Java memory switches in the **Java Options:** box.
4. Click **OK**. Restart the Tomcat service.

WebLogic 10.3.0

To update Java options for WebLogic 10.3.0:

1. Edit **setDomainEnv.cmd** in **BEA_HOME/user/projects/domains/*domain_name***.
2. Add arguments to the line:

   ```
   set MEM_ARGS=-Xms256m -Xmx512m
   ```

3. Run **setDomainEnv.cmd**.

Note: The `MEM_ARGS` parameter can also be updated in the startup script for the WebLogic domain.

Oracle WebCenter Standalone Products

Oracle WebCenter Analytics

To update Java options for the Oracle WebCenter Analytics JVM:

1. Edit **wrapper.conf** in **PT_HOME/ptanalytics/*version*/settings/config**.
2. Add or modify parameters in the section **Additional -D Java Properties**.

   Note: Java parameter numbers must be continuous and incremental, and are set in both **wrapper_base.conf** and **wrapper.conf**. Check both files to ensure added parameters use the next number in sequence.

Restart the Analytics service.

Automation Service

To update Java options for the Automation Service JVM:

1. Edit **wrapper.conf** in **PT_HOME/ptportal/*version*/settings/config**.
2. Add or modify parameters in the section **Additional -D Java Properties**.

   Note: Java parameter numbers must be continuous and incremental, and are set in both **wrapper_base.conf** and **wrapper.conf**. Check both files to ensure added parameters use the next number in sequence.

Restart the Automation service.

Oracle WebCenter Collaboration

To update Java options for the Oracle WebCenter Collaboration JVM:

1. Edit **wrapper.conf** in **PT_HOME/ptcollab/*version*/settings/config**.
2. Add or modify parameters in the section **Additional -D Java Properties**.

   Note: Java parameter numbers must be continuous and incremental, and are set in both **wrapper_base.conf** and **wrapper.conf**. Check both files to ensure added parameters use the next number in sequence.

3. Restart the Collaboration service.

Document Repository

To update Java options for the Document Repository JVM:

1. Edit **wrapper.conf** in **PT_HOME/ptdr/*version*/settings/config**.
2. Add or modify parameters in the section **Additional -D Java Properties**.

   Note: Java parameter numbers must be continuous and incremental, and are set in both **wrapper_base.conf** and **wrapper.conf**. Check both files to ensure added parameters use the next number in sequence.

Restart the Document Repository service.

Notification

To update Java options for the Notification JVM:

1. Edit **wrapper.conf** in **PT_HOME/ptnotification/*version*/settings/config**.
2. Add or modify parameters in the section **Additional -D Java Properties**.

   Note: Java parameter numbers must be continuous and incremental, and are set in both **wrapper_base.conf** and **wrapper.conf**. Check both files to ensure added parameters use the next number in sequence.

Restart the Notification service.

PTUpload

To update Java options for the PTUpload JVM:

1. Edit **wrapper.conf** in **PT_HOME/ptupload/*version*/settings/config**.
2. Add or modify parameters in the section **Additional -D Java Properties**.

   Note: Java parameter numbers must be continuous and incremental, and are set in both **wrapper_base.conf** and **wrapper.conf**. Check both files to ensure added parameters use the next number in sequence.

Restart the PTUpload service.

Oracle-BEA AquaLogic Interaction Publisher

To update Java options for the Publisher JVM:

1. Edit **service.conf** in **PT_HOME/ptcs/*version*/settings/config**.
2. Add a new parameter or modify existing parameters in the section **Java Additional Parameters**.

   For example, locate

```
# Java Additional Parameters
wrapper.java.additional.1=-Dprogram.name=cswfserver
wrapper.java.additional.2=-Djava.awt.headless=true
wrapper.java.additional.3=-Dplumtree.container.home=./../../../../../pt
cs/6.2/container
wrapper.java.additional.4=-Dplumtree.container.logs=./../../../../../pt
cs/6.2/logs
```

```
wrapper.java.additional.5=-Dorg.jboss.net.protocol.file.decodeFilePaths
=true
```

and add the garbage collection logging parameter

```
# Java Additional Parameters
wrapper.java.additional.1=-Dprogram.name=cswfserver
wrapper.java.additional.2=-Djava.awt.headless=true
wrapper.java.additional.3=-Dplumtree.container.home=./../../../../../pt
cs/6.2/container
wrapper.java.additional.4=-Dplumtree.container.logs=./../../../../../pt
cs/6.2/logs
wrapper.java.additional.5=-Dorg.jboss.net.protocol.file.decodeFilePaths
=true
wrapper.java.additional.6=-Xloggc:c:\publishergclog
```

3. Restart the Publisher service.

Note: Oracle-BEA AquaLogic Interaction Publisher and Oracle-BEA AquaLogic Interaction Workflow run on the same JVM. Garbage collection logging and memory tunings are for both services.

Oracle-BEA AquaLogic Interaction Studio

To update Java options for the Oracle-BEA AquaLogic Interaction Studio JVM:

1. Edit **wrapper.conf** in **PT_HOME/ptstudio/*version*/settings/config**.
2. Add or modify parameters in the section **Additional -D Java Properties**.

   Note: Java parameter numbers must be continuous and incremental, and are set in both **wrapper_base.conf** and **wrapper.conf**. Check both files to ensure added parameters use the next number in sequence.

Restart the Studio service.

Oracle-BEA AquaLogic Interaction Workflow

Oracle-BEA AquaLogic Interaction Workflow runs on the same JVM as Oracle-BEA AquaLogic Interaction Publisher and does not require separate tuning.