

Oracle® Process Manager and Notification Server
Administrator's Guide
10g (9.0.4)
Part No. B12057-02

March 2004

Contributors: Sanket Atal, Marcelo Goncalves, Kyle Jan, Alan Jones, Lypp-tek Khoo Ellis, Shaun Kopfenstein, Sharon Malek, Mukul Paithane, Udayini Pendyala, Jon Richards, Theresa Robertson, Mary Beth Roeser, Xilei (Kevin) Wang, Linjin Yu

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software--Restricted Rights (June 1987). Oracle Corporation, 500 Oracle Parkway, Redwood City, CA 94065

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

Contents

Send Us Your Comments	vii
Preface	ix
Audience.....	x
Documentation Accessibility	x
Organization	x
Related Documentation.....	xi
Conventions	xii
1 What's New in OPMN?	
1.1 Additional Oracle Application Server Component Management.....	1-2
1.2 New Commands	1-3
1.3 Start Order Dependencies.....	1-4
1.4 Event Scripts	1-4
1.5 Improved Monitoring.....	1-4
1.6 Operating System-Level Statistics	1-5
1.7 Changes to opmn.xml	1-5
1.8 DCM.....	1-6
2 OPMN: Overview	
2.1 What is OPMN?.....	2-2
2.2 How OPMN Works	2-3
2.2.1 Oracle Notification Server	2-3
2.2.2 Oracle Process Manager.....	2-4
2.2.3 PM Modules	2-4
2.3 What Oracle Application Server Components Does OPMN Manage?.....	2-5
2.3.1 Oracle Enterprise Manager 10g Application Server Control Console	2-5
2.4 OPMN Configuration Files.....	2-6
2.5 Automatic Restart	2-7
2.6 opmnctl Command Location.....	2-7
2.6.1 opmnctl Syntax.....	2-7
2.7 opmnctl Command Quick Reference.....	2-8
2.8 opmnctl Detailed Command Description	2-9
2.8.1 Server Control Commands.....	2-9
2.8.1.1 Server Control Commands on Microsoft Windows	2-9

2.8.1.2	opmnctl start	2-9
2.8.1.3	opmnctl startall	2-10
2.8.1.4	opmnctl stopall and opmnctl shutdown	2-10
2.8.1.5	opmnctl reload	2-11
2.8.2	Process Control Commands	2-11
2.8.2.1	opmnctl startproc, opmnctl restartproc and opmnctl stopproc	2-11
2.8.3	Command Definitions	2-13
2.8.3.1	Scope	2-13
2.8.3.2	Attributes	2-14
2.8.3.3	Verbose	2-15
2.8.3.4	Options	2-16
2.8.4	Status Commands	2-18
2.8.4.1	opmnctl status	2-18
2.8.4.2	opmnctl ping	2-19
2.8.5	Help Commands	2-19
2.8.5.1	opmnctl help	2-19
2.8.5.2	opmnctl usage	2-20
2.8.5.3	opmnctl validate	2-21
2.9	OPMN Log Files	2-22
2.10	Security	2-22
2.10.1	Remote Security	2-23
2.11	iHAT	2-23

3 Using OPMN

3.1	Starting OPMN	3-2
3.2	Starting and Stopping OPMN Managed Processes for a Local Oracle Application Server Instance	3-2
3.3	Starting and Stopping all OPMN Managed Processes for a Remote Oracle Application Server Instance	3-2
3.4	Starting and Stopping an Oracle Application Server Component in a Local Oracle Application Server Instance	3-2
3.5	Starting and Stopping an Oracle Application Server Process Type in a Local Oracle Application Server Instance	3-3
3.6	Starting and Stopping a Multi-Oracle Application Server Instance Environment	3-3
3.7	Starting a Component on an Oracle Application Server Farm	3-3

4 opmn.xml Common Configuration

4.1	Example of opmn.xml Elements and Attributes	4-2
4.2	opmn.xml Element and Attribute Descriptions	4-4

5 Configuring Oracle HTTP Server

5.1	Oracle HTTP Server Process Module Configuration	5-2
5.2	Oracle HTTP Server Minimum Configuration	5-2
5.3	Oracle HTTP Server Complete Configuration	5-2
5.4	Oracle HTTP Server Attribute Descriptions	5-3

6 Configuring OC4J

6.1	OC4J Process Module Configuration	6-2
6.2	OC4J Minimum Configuration	6-2
6.3	OC4J Complete Configuration	6-2
6.4	OC4J Attribute Descriptions.....	6-3

7 Configuring Oracle Application Server Web Cache

7.1	OracleAS Web Cache Process Module Configuration	7-2
7.2	OracleAS Web Cache Minimum Configuration.....	7-2
7.3	OracleAS Web Cache Complete Configuration	7-2
7.4	OracleAS Web Cache Attribute Descriptions	7-3

8 Configuring Oracle Internet Directory

8.1	Oracle Internet Directory Process Module Configuration	8-2
8.2	Oracle Internet Directory Minimum Configuration.....	8-2
8.3	Oracle Internet Directory Complete Configuration.....	8-3
8.4	Oracle Internet Directory Attribute Descriptions	8-4

9 Configuring Oracle Application Server Port Tunnel

9.1	OracleAS Port Tunnel Process Module Configuration	9-2
9.2	OracleAS Port Tunnel Minimum Configuration.....	9-2
9.3	OracleAS Port Tunnel Complete Configuration	9-2
9.4	OracleAS Port Tunnel Attribute Descriptions	9-2

10 Configuring Oracle Application Server Wireless

10.1	OracleAS Wireless Process Module Configuration	10-2
10.2	OracleAS Wireless Minimum Configuration.....	10-2
10.3	OracleAS Wireless Complete Configuration	10-3
10.4	OracleAS Wireless Attribute Descriptions	10-6

11 Configuring Oracle Application Server ProcessConnect

11.1	OracleAS ProcessConnect Module Configuration.....	11-2
11.2	OracleAS ProcessConnect Minimum Configuration.....	11-2
11.3	OracleAS ProcessConnect Complete Configuration	11-3
11.4	OracleAS ProcessConnect Attribute Descriptions	11-4

12 Configuring Oracle Application Server Reports Services

12.1	Introduction	12-2
12.2	OracleAS Reports Services Process Module Configuration	12-2
12.3	OracleAS Reports Services Minimum Configuration.....	12-2
12.3.1	In-process OracleAS Reports Services Server.....	12-2
12.3.2	Standalone OracleAS Reports Services Server	12-3
12.4	OracleAS Reports Services Complete Configuration	12-3
12.4.1	In-process OracleAS Reports Services Server.....	12-3

12.4.2	Standalone OracleAS Reports Services Server	12-4
12.5	OracleAS Reports Services Attribute Descriptions.....	12-4

13 Configuring Oracle Application Server Discoverer

13.1	OracleAS Discoverer Process Module Configuration	13-2
13.2	OracleAS Discoverer Minimum Configuration.....	13-2
13.3	OracleAS Discoverer Complete Configuration	13-3
13.4	OracleAS Discoverer Attribute Descriptions	13-4

14 Configuring Log Loader

14.1	Log Loader Process Module Configuration	14-2
14.2	Log Loader Minimum Configuration.....	14-2
14.3	Log Loader Complete Configuration	14-3
14.4	Log Loader Attribute Descriptions	14-3

15 Configuring DCM Daemon

15.1	DCM Daemon Process Module Configuration	15-2
15.2	DCM Daemon Minimum Configuration.....	15-2
15.3	DCM Daemon Complete Configuration	15-2
15.4	DCM Daemon Attribute Descriptions	15-3

16 Configuring Custom Process

16.1	Custom Process Module Configuration	16-2
16.2	Custom Process Minimum Configuration.....	16-2
16.3	Custom Process Complete Configuration	16-2
16.3.1	Ping	16-2
16.4	Custom Process Attribute Descriptions.....	16-4

17 OPMN Troubleshooting

17.1	Diagnosing OPMN Problems.....	17-2
17.1.1	OPMN log Files.....	17-2
17.1.2	opmnctl debug	17-2
17.1.3	iHAT	17-3
17.1.4	Oracle Enterprise Manager 10g Application Server Control Console	17-3
17.1.5	Troubleshooting with Event Scripts.....	17-4
17.2	Common Problems	17-5
17.2.1	Oracle Application Server Process Does Not Start	17-5
17.2.2	Determining if Oracle Application Server Processes are Dying or Unresponsive..	17-6
17.2.3	opmnctl Command Execution Times Out.....	17-6
17.2.4	Oracle Application Server Component Automatically Restarted by OPMN	17-7
17.2.5	opmn.xml Environment Variables	17-7
17.2.6	Unexpected opmnctl start Behavior.....	17-7
17.2.7	Disabled Element in the opmn.xml File	17-7
17.2.8	Management of Oracle9iAS Release 2 (9.0.2 and 9.0.3) Instances	17-8

Index

Send Us Your Comments

**Oracle Process Manager and Notification Server Administrator's Guide, 10g
(9.0.4)**

Part No. B12057-02

Oracle welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most about this manual?

If you find any errors or have any other suggestions for improvement, please indicate the title and part number of the documentation and the chapter, section, and page number (if available). You can send comments to us in the following ways:

- Electronic mail: appserverdocs_us@oracle.com
- FAX: 650-506-7365 Attn: Oracle Application Server Documentation Manager
- Postal service:
Oracle Corporation
Oracle Application Server Documentation
500 Oracle Parkway, M/S 1op6
Redwood Shores, CA 94065
USA

If you would like a reply, please give your name, address, telephone number, and electronic mail address (optional).

If you have problems with the software, please contact your local Oracle Support Services.

Preface

This guide describes how to administer Oracle Process Manager and Notification Server (OPMN) in order to manage Oracle Application Server components.

This preface contains these topics:

- [Audience](#)
- [Documentation Accessibility](#)
- [Organization](#)
- [Related Documentation](#)
- [Conventions](#)

Audience

The Oracle Process Manager and Notification Server Administrator's Guide is intended for administrators of Oracle Application Server.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For additional information, visit the Oracle Accessibility Program Web site at

<http://www.oracle.com/accessibility/>

Accessibility of Code Examples in Documentation JAWS, a Windows screen reader, may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, JAWS may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Organization

This document contains:

Chapter 1, "What's New in OPMN?"

This chapter describes new OPMN features available in Oracle Application Server 10g (9.0.4).

Chapter 2, "OPMN: Overview"

This chapter provides an overview of the OPMN.

Chapter 3, "Using OPMN"

This chapter provides command-line examples on how to use OPMN.

Chapter 4, "opmn.xml Common Configuration"

This chapter provides common configuration examples, and descriptions of elements and attributes for the `opmn.xml` file.

Chapter 5, "Configuring Oracle HTTP Server"

This chapter describes Oracle HTTP Server configuration in the `opmn.xml` file.

Chapter 6, "Configuring OC4J"

This chapter describes OC4J configuration in the `opmn.xml` file.

Chapter 7, "Configuring Oracle Application Server Web Cache"

This chapter describes OracleAS Web Cache configuration in the `opmn.xml` file.

Chapter 8, "Configuring Oracle Internet Directory"

This chapter describes Oracle Internet Directory configuration in the `opmn.xml` file.

Chapter 9, "Configuring Oracle Application Server Port Tunnel"

This chapter describes Oracle Application Server Port Tunnel configuration in the `opmn.xml` file.

Chapter 10, "Configuring Oracle Application Server Wireless"

This chapter describes OracleAS Wireless configuration in the `opmn.xml` file.

Chapter 11, "Configuring Oracle Application Server ProcessConnect"

This chapter describes OracleAS ProcessConnect configuration in the `opmn.xml` file.

Chapter 12, "Configuring Oracle Application Server Reports Services"

This chapter describes OracleAS Reports Services configuration in the `opmn.xml` file.

Chapter 13, "Configuring Oracle Application Server Discoverer"

This chapter describes OracleAS Discoverer configuration in the `opmn.xml` file.

Chapter 14, "Configuring Log Loader"

This chapter describes Log Loader configuration in the `opmn.xml` file.

Chapter 15, "Configuring DCM Daemon"

This chapter describes DCM Daemon configuration in the `opmn.xml` file.

Chapter 16, "Configuring Custom Process"

This chapter describes custom process configuration in the `opmn.xml` file.

Chapter 17, "OPMN Troubleshooting"

This chapter describes some troubleshooting tips for OPMN.

Related Documentation

For more information, see these Oracle resources:

- Oracle Application Server Documentation Library
- Oracle Application Server Platform-Specific Documentation on Oracle Application Server Disk 1

Printed documentation is available for sale in the Oracle Store at

<http://oraclestore.oracle.com/>

To download free release notes, installation documentation, white papers, or other collateral, please visit the Oracle Technology Network (OTN). You must register online before using OTN; registration is free and can be done at

<http://otn.oracle.com/membership/>

If you already have a username and password for OTN, then you can go directly to the documentation section of the OTN Web site at

<http://otn.oracle.com/documentation/>

Conventions

This section describes the conventions used in the text and code examples of this documentation set. It describes:

- [Conventions in Text](#)
- [Conventions in Code Examples](#)
- [Conventions for Microsoft Windows Operating Systems](#)

Conventions in Text

We use various conventions in text to help you more quickly identify special terms. The following table describes those conventions and provides examples of their use.

Convention	Meaning	Example
Bold	Bold typeface indicates terms that are defined in the text or terms that appear in a glossary, or both.	When you specify this clause, you create an index-organized table .
<i>Italics</i>	Italic typeface indicates book titles or emphasis.	<i>Oracle9i Database Concepts</i> Ensure that the recovery catalog and target database do <i>not</i> reside on the same disk.
UPPERCASE monospace (fixed-width) font	Uppercase monospace typeface indicates elements supplied by the system. Such elements include parameters, privileges, datatypes, RMAN keywords, SQL keywords, SQL*Plus or utility commands, packages and methods, as well as system-supplied column names, database objects and structures, usernames, and roles.	You can specify this clause only for a NUMBER column. You can back up the database by using the BACKUP command. Query the TABLE_NAME column in the USER_TABLES data dictionary view. Use the DBMS_STATS.GENERATE_STATS procedure.
lowercase monospace (fixed-width) font	Lowercase monospace typeface indicates executables, filenames, directory names, and sample user-supplied elements. Such elements include computer and database names, net service names, and connect identifiers, as well as user-supplied database objects and structures, column names, packages and classes, usernames and roles, program units, and parameter values. Note: Some programmatic elements use a mixture of UPPERCASE and lowercase. Enter these elements as shown.	Enter sqlplus to open SQL*Plus. The password is specified in the orapwd file. Back up the datafiles and control files in the /disk1/oracle/dbs directory. The department_id, department_name, and location_id columns are in the hr.departments table. Set the QUERY_REWRITE_ENABLED initialization parameter to true. Connect as oe user. The JRepUtil class implements these methods.
<i>lowercase italic monospace (fixed-width) font</i>	Lowercase italic monospace font represents placeholders or variables.	You can specify the <i>parallel_clause</i> . Run <i>Old_release.SQL</i> where <i>old_release</i> refers to the release you installed prior to upgrading.

Conventions in Code Examples

Code examples illustrate SQL, PL/SQL, SQL*Plus, or other command-line statements. They are displayed in a monospace (fixed-width) font and separated from normal text as shown in this example:

```
SELECT username FROM dba_users WHERE username = 'MIGRATE';
```

The following table describes typographic conventions used in code examples and provides examples of their use.

Convention	Meaning	Example
[]	Brackets enclose one or more optional items. Do not enter the brackets.	DECIMAL (<i>digits</i> [, <i>precision</i>])
{ }	Braces enclose two or more items, one of which is required. Do not enter the braces.	{ENABLE DISABLE}
	A vertical bar represents a choice of two or more options within brackets or braces. Enter one of the options. Do not enter the vertical bar.	{ENABLE DISABLE} [COMPRESS NOCOMPRESS]
...	Horizontal ellipsis points indicate either: <ul style="list-style-type: none">■ That we have omitted parts of the code that are not directly related to the example■ That you can repeat a portion of the code	CREATE TABLE ... AS <i>subquery</i> ; SELECT <i>col1</i> , <i>col2</i> , ... , <i>coln</i> FROM employees;
.	Vertical ellipsis points indicate that we have omitted several lines of code not directly related to the example.	
.		
.		
Other notation	You must enter symbols other than brackets, braces, vertical bars, and ellipsis points as shown.	acctbal NUMBER(11,2); acct CONSTANT NUMBER(4) := 3;
<i>Italics</i>	Italicized text indicates placeholders or variables for which you must supply particular values.	CONNECT SYSTEM/ <i>system_password</i> DB_NAME = <i>database_name</i>
UPPERCASE	Uppercase typeface indicates elements supplied by the system. We show these terms in uppercase in order to distinguish them from terms you define. Unless terms appear in brackets, enter them in the order and with the spelling shown. However, because these terms are not case sensitive, you can enter them in lowercase.	SELECT last_name, employee_id FROM employees; SELECT * FROM USER_TABLES; DROP TABLE hr.employees;
lowercase	Lowercase typeface indicates programmatic elements that you supply. For example, lowercase indicates names of tables, columns, or files. Note: Some programmatic elements use a mixture of UPPERCASE and lowercase. Enter these elements as shown.	SELECT last_name, employee_id FROM employees; sqlplus hr/hr CREATE USER mjones IDENTIFIED BY ty3MU9;

Conventions for Microsoft Windows Operating Systems

The following table describes conventions for Microsoft Windows operating systems and provides examples of their use.

Convention	Meaning	Example
Choose Start >	How to start a program.	To start the Oracle Database Configuration Assistant, choose Start > Programs > Oracle - <i>HOME_NAME</i> > Configuration and Migration Tools > Database Configuration Assistant.
File and directory names	File and directory names are not case sensitive. The following special characters are not allowed: left angle bracket (<), right angle bracket (>), colon (:), double quotation marks ("), slash (/), pipe (), and dash (-). The special character backslash (\) is treated as an element separator, even when it appears in quotes. If the file name begins with \\, then Windows assumes it uses the Universal Naming Convention.	c:\winnt \"system32 is the same as C:\WINNT\SYSTEM32
C :\>	Represents the Windows command prompt of the current hard disk drive. The escape character in a command prompt is the caret (^). Your prompt reflects the subdirectory in which you are working. Referred to as the <i>command prompt</i> in this manual.	C:\oracle\oradata>
	The backslash (\) special character is sometimes required as an escape character for the double quotation mark ("') special character at the Windows command prompt. Parentheses and the single quotation mark ('') do not require an escape character. Refer to your Windows operating system documentation for more information on escape and special characters.	C:\>exp scott/tiger TABLES=emp QUERY=\ "WHERE job='SALESMAN' and sal<1600\" C:\>imp SYSTEM/password FROMUSER=scott TABLES=(emp, dept)
<i>HOME_NAME</i>	Represents the Oracle home name. The home name can be up to 16 alphanumeric characters. The only special character allowed in the home name is the underscore.	C:\> net start Oracle <i>HOME_NAME</i> TNSListener

Convention	Meaning	Example
<i>ORACLE_HOME</i> and <i>ORACLE_BASE</i>	<p>In releases prior to Oracle8i release 8.1.3, when you installed Oracle components, all subdirectories were located under a top level <i>ORACLE_HOME</i> directory that by default used one of the following names:</p> <ul style="list-style-type: none"> ■ C:\orant for Windows NT ■ C:\orawin95 for Windows 95 ■ C:\orawin98 for Windows 98 <p>This release complies with Optimal Flexible Architecture (OFA) guidelines. All subdirectories are not under a top level <i>ORACLE_HOME</i> directory. There is a top level directory called <i>ORACLE_BASE</i> that by default is C:\oracle. If you install Oracle9<i>i</i> release 1 (9.0.1) on a computer with no other Oracle software installed, then the default setting for the first Oracle home directory is C:\oracle\ora90. The Oracle home directory is located directly under <i>ORACLE_BASE</i>.</p> <p>All directory path examples in this guide follow OFA conventions.</p> <p>Refer to <i>Oracle9i Database Getting Starting for Windows</i> for additional information about OFA compliances and for information about installing Oracle products in non-OFA compliant directories.</p>	Go to the <i>ORACLE_BASE\ORACLE_HOME\rdbms\admin</i> directory.

What's New in OPMN?

This chapter describes the new features of OPMN in Oracle Application Server. OPMN allows you to manage Oracle Application Server components.

This chapter includes the following topics:

- [Additional Oracle Application Server Component Management](#)
- [New Commands](#)
- [Start Order Dependencies](#)
- [Event Scripts](#)
- [Improved Monitoring](#)
- [Operating System-Level Statistics](#)
- [Changes to opmn.xml](#)
- [DCM](#)

1.1 Additional Oracle Application Server Component Management

In 10g (9.0.4), OPMN allows you to manage additional Oracle Application Server components when compared to Oracle9*i* Application Server (Oracle9*i*AS) Release 2 (9.0.2 and 9.0.3).

[Table 1-1](#) shows the additional Oracle Application Server components managed by OPMN in 10g (9.0.4):

Table 1-1 Oracle Application Server Component Management

Oracle Application Server Component	OPMN Management	Release 2 (9.0.2 and 9.0.3)	10g (9.0.4)
Oracle HTTP Server	YES	YES	YES
Oracle Application Server Containers for J2EE (OC4J)	YES	YES	YES
Distributed Configuration Management (DCM) daemon (server)			YES
OracleAS Log Loader			YES
Oracle Internet Directory			YES
OracleAS Port Tunnel			YES
OracleAS Web Cache			YES
OracleAS Discoverer			YES
OracleAS Wireless			YES
OracleAS Reports Services			YES
OracleAS ProcessConnect			YES

OPMN explicitly manages the listed 10g (9.0.4) components. OPMN also manages other Oracle Application Server components implicitly; the constituent parts of the implicit Oracle Application Server components are managed by OPMN as part of one or more other Oracle Application Server components. For example, OracleAS Portal while not explicitly managed by OPMN is implicitly managed because it is operational using OC4J and Oracle HTTP Server.

The following components are implicitly managed by OPMN:

- Oracle Business Components for Java (BC4J)
- OracleAS Forms Services
- OracleAS Single Sign-On
- Oracle HTTP Server Adapter of OracleAS InterConnect
- Oracle Distributed Authoring and Versioning (OraDAV)
- Oracle Personalization Server
- OracleAS Portal
- OracleAS SOAP
- Oracle Syndication Server
- OracleAS UDDI Registry Server
- Oracle Ultra Search
- OracleAS TopLink

1.2 New Commands

`opmnctl` is the recommended command-line tool for starting and stopping Oracle Application Server components. `opmnctl` is the centralized way to control and monitor Oracle Application Server components from the command line. You can use `opmnctl` to execute control and monitoring commands across multiple Oracle Application Server instances simultaneously.

For Oracle9iAS Release 2 (9.0.2 and 9.0.3), command-line start and stop was accomplished using `dcmctl` and a number of component-specific tools, such as `webcachectl` and `oidctl`.

In Oracle Application Server 10g (9.0.4), `opmnctl` is the supported tool for starting and stopping all components in an Oracle Application Server instance, with the exception of the Oracle Application Server Metadata Repository and the Oracle Enterprise Manager 10g Application Server Control Console (Application Server Control Console). `opmnctl` also allows you to perform operations on a specified Oracle Application Server instance on the application server farm, all instances in the farm, and all instances in a cluster using an optional parameter called `scope`. You can also use the `scope` option to control an individual Oracle Application Server process.

See Also:

- [Oracle Application Server 10g Administrator's Guide](#)
- [Chapter 2, "OPMN: Overview"](#)
- [Chapter 3, "Using OPMN"](#)

1.3 Start Order Dependencies

Some Oracle Application Server components and services require that other components and services are up and running before starting. OPMN is configured at installation with default start order dependencies, which allows you to start all of the components in an instance in the proper order with a single command. Refer to the *Oracle Application Server 10g Administrator's Guide* for more information on Oracle Application Server dependencies.

OPMN is configured with a set of dependencies but you can configure additional dependencies according to the environment

See Also: [Chapter 2, "OPMN: Overview"](#)

1.4 Event Scripts

You can configure OPMN to execute your own custom event scripts whenever a particular component starts, stops, or crashes. You can select from one or more of the following event types:

- **pre-start:** OPMN runs the pre-start script after any configured dependency checks have been performed and passed, and before the Oracle Application Server component starts. For example, the pre-start script can be used for site-specific initialization of external components.
- **pre-stop:** OPMN runs the pre-stop script before stopping a designated Oracle Application Server component. For example, the pre-stop script can be used for collecting Java Virtual Machine stack traces prior to stopping OC4J processes.
- **post-crash:** OPMN runs the post-crash script after the Oracle Application Server component has terminated unexpectedly. For example, a user could learn of component crashes by supplying a script or program to be executed at post-crash events which sends a notification to the administrator's pager."

See Also: [Chapter 2, "OPMN: Overview"](#)

1.5 Improved Monitoring

For Oracle9iAS Release 2 (9.0.2 and 9.0.3), OPMN provided several undocumented commands for obtaining information from OPMN about the Oracle Application Server components and processes it managed. However, the obtained information did not provide enough detail about the OPMN processes. Additionally, the obtained information was not in a standard format.

In Oracle Application Server 10g (9.0.4), OPMN has consolidated all the Release 2 (9.0.2 and 9.0.3) commands into a single command that provides more flexibility and returns information in a choice of standard formats.

The following are some examples of the process level information that can be obtained from OPMN:

- status information about multiple Oracle Application Server instances with a single command
- information on CPU and memory utilization
- what ports are in use by a process (for those Oracle Application Server components that can report them)
- operating system-level process identifiers (pid)
- the Oracle Application Server component to which a process belongs
- whether a process is up, down, or initializing

See Also: [Chapter 2, "OPMN: Overview"](#)

1.6 Operating System-Level Statistics

The Application Server Control Console reports CPU and memory utilization information for all of Oracle Application Server.

For Oracle9iAS Release 2 (9.0.2 and 9.03), Oracle Enterprise Manager 10g obtained CPU and memory utilization statistics by identifying running components at the operating system level. The internal operating interfaces for obtaining such utilization statistics vary greatly from one operating system to another. Additionally, there was no easy and efficient way to do this within Java environment.

In Oracle Application Server 10g (9.0.4), OPMN tracks CPU and memory statistics and stores the information for all Oracle Application Server components directly launched by OPMN. Application Server Control Console obtains this information from OPMN by sending requests. OPMN obtains the operating system level process statistics in a very efficient manner thereby improving the time it takes to load Application Server Control Console pages.

1.7 Changes to opmn.xml

The `opmn.xml` file in Oracle Application Server 10g (9.0.4) has changed when compared to the previous Oracle9iAS Release 2 (9.0.2 and 9.0.3) versions.

For Oracle9iAS Release 2 (9.0.2 and 9.0.3), the `opmn.xml` file contained XML element names specific to each Oracle Application Server component; an Oracle HTTP Server element encloses the Oracle HTTP Server configuration, and an OC4J element encloses the OC4J configuration. Because of this requirement, Oracle9iAS Release 2 (9.0.2 and 9.0.3) `opmn.xml` lacked flexibility. Adding management of new Oracle9iAS components required changes to the XML schema and changes to the configuration processing code to look for the new elements.

In Oracle Application Server 10g (9.0.4), all component-specific element names have been removed. In addition, all component specific management code has been moved into Process Management (PM) modules which get loaded by OPMN at startup according to what has been specified in the **modules** section of `opmn.xml`.

Each level has a specific set of configurations. In addition, there are several configuration elements that are accepted at more than one level to provide the flexibility of applying a configuration across an entire Oracle Application Server component or just part of a component.

1.8 DCM

DCM manages the configuration of `opmn.xml` and manages configurations among application server instances that are associated with a common Infrastructure (members of an Oracle Application Server farm). It enables Oracle Application Server cluster-wide deployment so you can deploy an application to an entire cluster, or make a single host or instance configuration change applicable across all instances in a cluster. The Application Server Control Console uses DCM to make configuration changes and to propagate configuration changes and deployed applications across the cluster.

All command line process control should be through the OPMN `opmnctl` command. The DCM `dcmctl` command should be used only for configuration related operations and application deployment.

See Also: *Oracle Application Server 10g Administrator's Guide*

2

OPMN: Overview

This chapter provides an overview of OPMN for Oracle Application Server. It features the following topics:

- [What is OPMN?](#)
- [How OPMN Works](#)
- [What Oracle Application Server Components Does OPMN Manage?](#)
- [OPMN Configuration Files](#)
- [opmnctl Command Location](#)
- [opmnctl Command Quick Reference](#)
- [opmnctl Detailed Command Description](#)
- [OPMN Log Files](#)
- [Security](#)
- [iHAT](#)

2.1 What is OPMN?

OPMN is installed and configured with every Oracle Application Server installation type and is essential for running Oracle Application Server.

OPMN features the following functionality:

- Provides a command-line interface for process control and monitoring for single or multiple Oracle Application Server components and instances.
- Provides an integrated way to manage Oracle Application Server components.
- Enables management of Oracle Application Server subcomponents and sub-subcomponents.
- Channels all events from different Oracle Application Server component instances to all Oracle Application Server components that can utilize them.
- Solves interdependency issues between Oracle Application Server components by enabling you to start and stop components in order.
- Enables customizing of enterprise functionality by using event scripts.
- Enables gathering of host and Oracle Application Server process statistics and tasks.
- Provides automatic restart of Oracle Application Server processes when they become unresponsive, terminate unexpectedly, or become unreachable as determined by ping and notification operations.
- Provides automatic death detection of Oracle Application Server processes.
- Does not depend on any other Oracle Application Server component being up and running before it can be started and used.

The OPMN server should be started as soon as possible after turning on the host. OPMN must be running whenever OPMN-managed components are turned on or off.

Note: On the Microsoft Windows operating system, OPMN is installed as a Windows service (`Oracle<OracleHomename>ProcessManager`). It starts up automatically when you start or reboot your computer. Refer to [Section 2.8.1.3, "opmnctl startall"](#) for more information.

Oracle Application Server components managed by OPMN should never be started or stopped manually. Do not use command line scripts or utilities from previous versions of Oracle Application Server for starting and stopping Oracle Application Server components. OPMN must be the last service turned off whenever you reboot or turn off your computer.

Use the Application Server Control Console and the `opmnctl` command line utility to start or stop Oracle Application Server components.

2.2 How OPMN Works

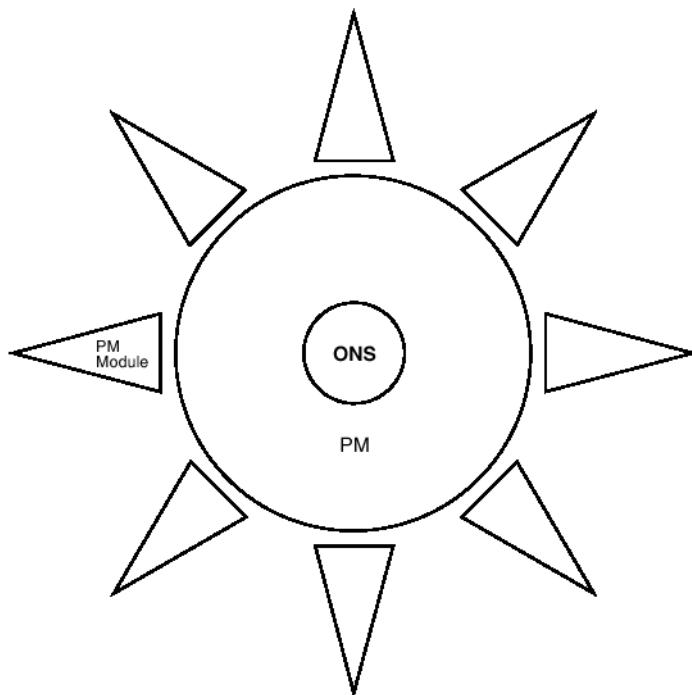
OPMN consists of a core grouping of three components that interpret and convey notification information sent between Oracle Application Server processes within the same or different OPMN servers.

The core of OPMN consists of the following three components:

- [Oracle Notification Server](#)
- [Oracle Process Manager](#)
- [PM Modules](#)

[Figure 2–1](#) shows the architecture of the core of OPMN.

Figure 2–1 OPMN Architecture



2.2.1 Oracle Notification Server

Oracle Notification Server (ONS) is the transport mechanism for failure, recovery, startup, and other related notifications between components in Oracle Application Server. It operates according to a publish-subscribe model: an Oracle Application Server component receives a notification of a certain type per its subscription to ONS. When such a notification is published, ONS sends it to the appropriate subscribers.

2.2.2 Oracle Process Manager

Oracle Process Manager (PM) is the centralized process management mechanism in Oracle Application Server and is used to manage Oracle Application Server processes. It starts, stops, restarts, and detects death of these processes. The Oracle Application Server processes that PM is configured to manage are specified in the `opmn.xml` file.

The PM waits for a user command to start specific, or all processes. When a specific process or all processes are to be stopped, the PM receives a request as specified by the request parameters. OPMN is monitored by a shadow process that restarts upon request or after a catastrophic failure.

The PM uses the ONS to:

- detect that a process has completed initialization and is ready to receive requests
- determine what ports are in use
- obtain component specific runtime information

The Application Server Control Console also uses PM to manage processes.

2.2.3 PM Modules

The Oracle Process Manager Modules (PM Modules) implement Oracle Application Server component-specific process management functionality. The PM Modules pass notification information returned by other Oracle Application Server component PM Modules within the same or different OPMN servers.

The PM Modules:

- the Oracle Application Server component specific notification information is sent from running component processes.
- construct Oracle Application Server component specific control information.
- test responsiveness in an Oracle Application Server component specific manner to determine if a component is responding to requests.

2.3 What Oracle Application Server Components Does OPMN Manage?

OPMN manages all Oracle Application Server components except the OracleAS Metadata Repository or the Application Server Control Console.

You can also configure OPMN to manage other components; for example, OracleAS Port Tunnel, and custom processes.

Because of the extensible design of OPMN, add-on components are managed by OPMN without having to update OPMN itself.

See Also:

- [Chapter 9, "Configuring Oracle Application Server Port Tunnel"](#)
- [Chapter 16, "Configuring Custom Process"](#)

2.3.1 Oracle Enterprise Manager 10g Application Server Control Console

In addition to OPMN, you can also manage your enterprise using the Application Server Control Console. Application Server Control Console leverages the functionality of OPMN to manage your Oracle Application Server enterprise. Using a Web browser, Application Server Control Console provides a graphical interface that enables management of all Oracle Application Server components in your network and enterprise.

See Also: *Oracle Application Server 10g Administrator's Guide*

2.4 OPMN Configuration Files

OPMN reads the following configuration files when started or reloaded:

- `ORACLE_HOME/opmn/conf/opmn.xml`

The `opmn.xml` file is the main configuration file for OPMN. The `opmn.xml` file contains information for the ONS, the PM, and Oracle Application Server component specific configuration. The `opmn.xml` file shows you which Oracle Application Server components OPMN is managing on your system.

The `opmn.xml` file contains Oracle Application Server component entries arranged in the following hierarchical structure:

```
<ias-component>
  <process-type>
    <process-set>
```

`<ias-component>`: This entry represents the Oracle Application Server component. It enables management of the component for processes such as starting and stopping.

`<process-type>`: This subcomponent of the `<ias-component>` entry declares the type of process to run by association with a specific PM module.

`<process-set>`: This sub-subcomponent of the `<ias-component>` entry enables you to declare different sets of optional runtime arguments and environments for the Oracle Application Server component.

The following [Example 2-1](#) shows an example of the `<ias-component>`, `<process-type>`, `<process-set>` entries in an `opmn.xml` file.

Example 2-1 opmn.xml file

```
<ias-component id="OC4J">
  <process-type id="home">
    <process-set id="default_island">....
```

You can edit `opmn.xml` using Application Server Control Console. Click the **Process Management** link at the bottom of the Oracle Application Server instance home page. Do not stop the OPMN server after you edit the `opmn.xml` file. Application Server Control Console automatically reloads the updated `opmn.xml` file after you edit the file.

If you manually edit the `opmn.xml` file run the `dcmctl updateConfig` command on the command line. `dcmctl updateConfig` reloads the updated file and updates the configuration repository with the manual changes.

- `ORACLE_HOME/opmn/conf/ons.conf`

OPMN uses the `ons.conf` file to find all the different Oracle Application Server instances in the farm. The `ons.conf` file is automatically generated and maintained by DCM. Do not edit the `ons.conf` file; edits to this file are overwritten by DCM.

- `ORACLE_HOME/dcm/config/dcm.conf`

The `dcm.conf` file specifies the instance name and cluster name of the local Oracle Application Server installation. The `dcm.conf` file is automatically generated and maintained by DCM. The `dcm.conf` file should not be edited manually.

2.5 Automatic Restart

OPMN gives the user control over automatic death detection and restart of components; you can configure the parameters by which OPMN determines a process has died and disable automatic restart for individual components.

OPMN monitors the operation of its managed processes by the following methods:

- Operating system level detection of Oracle Application Server process death
- Periodic ping requests to Oracle Application Server processes
- Periodic status notification from Oracle Application Server processes ("reverse-ping")

The ping and notification functionality is only used where appropriate according to the functionality of the Oracle Application Server component.

OPMN automatically restarts Oracle Application Server components that terminate unexpectedly. OPMN will also restart processes that are unresponsive according to the result of notification and ping operations.

See Also: [Chapter 4, "opmn.xml Common Configuration"](#)

2.6 opmnctl Command Location

opmnctl is the command-line utility for OPMN. The opmnctl command is located in the following directory locations:

(UNIX) *ORACLE_HOME*/opmn/bin/opmnctl

(Windows) *ORACLE_HOME*\opmn\bin\opmnctl

Note: Oracle Application Server components managed by OPMN should never be started or stopped manually. Do not use command line scripts or utilities from previous versions of Oracle Application Server for starting and stopping Oracle Application Server components. Use the Application Server Control Console and the opmnctl command line utility to start or stop Oracle Application Server components.

Note: Oracle recommends starting OPMN as the user that has installed Oracle Application Server.

2.6.1 opmnctl Syntax

The command below shows an example of the syntax of the opmnctl command:

```
opmnctl [verbose] [<scope>] <command> [<options>]
```

[Table 2–1](#) provides a description about opmnctl syntax.

Table 2–1 opmnctl Syntax

Syntax	Description
verbose	Prints detailed execution message, if available.
scope	Specifies where the request is routed. Refer to Section 2.8.3.1, "Scope" for a list of options.

Table 2–1 (Cont.) opmnctl Syntax

Syntax	Description
command	Specifies an opmnctl command. Refer to Example 2–2 for a list of commands.
options	Specifies options for the command. Refer to Section 2.8.3.4, "Options" for a list of options.

2.7 opmnctl Command Quick Reference

[Example 2–2](#) lists opmnctl commands for quick reference. You can obtain the same output information by executing the opmnctl help command.

Example 2–2 opmnctl Commands

```
prompt >  
opmnctl help
```

scope	command	options
	start	- Start opmn
	startall	- Start opmn and all managed processes
	stopall	- Stop opmn and all managed processes
	shutdown	- Shutdown opmn and all managed processes
[<scope>]	startproc [<attr>=<val>..]	- Start opmn managed processes
[<scope>]	restartproc [<attr>=<val>..]	- Restart opmn managed processes
[<scope>]	stopproc [<attr>=<val>..]	- Stop opmn managed processes
[<scope>]	reload	- Trigger opmn to reread opmn.xml
[<scope>]	status [<options>]	- Get managed process status
	ping [<max_retry>]	- Ping local opmn
	validate [<filename>]	- Validate the given xml file
	help	- Print brief usage description
	usage [<command>]	- Print detailed usage description

2.8 opmnctl Detailed Command Description

The following sections contains detailed descriptions of the opmnctl commands listed in [Example 2–2](#). The opmnctl commands are displayed in the following sections:

- [Server Control Commands](#)
- [Process Control Commands](#)
- [Command Definitions](#)
- [Status Commands](#)
- [Help Commands](#)

2.8.1 Server Control Commands

The opmnctl start, startall, reload, stopall, and shutdown commands enable you to control the OPMN server.

- [opmnctl start](#)
- [opmnctl startall](#)
- [opmnctl stopall and opmnctl shutdown](#)
- [opmnctl reload](#)

Output is not generated for the successful execution of an opmnctl server control command. Refer to [Chapter 17, "OPMN Troubleshooting"](#) if you receive any error messages during opmnctl command execution.

2.8.1.1 Server Control Commands on Microsoft Windows

On the Microsoft Windows operating system, OPMN is installed as a Windows service (`Oracle<OracleHomename>ProcessManager`) and it starts up automatically when you reboot your computer. When you start or stop OPMN using Windows Services you start or stop *all* OPMN managed components on the local instance.

Use the Application Server Control Console and the opmnctl command line utility to start or stop Oracle Application Server components.

2.8.1.2 opmnctl start

Syntax: `opmnctl start`

Use this command to start the OPMN server for a local Oracle Application Server instance without starting OPMN-managed processes.

Execute this command as soon as possible after starting your computer.

OPMN starts up automatically on Microsoft Windows when you start or reboot your computer. All OPMN-managed processes are started.

See Also: [Chapter 3, "Using OPMN"](#)

2.8.1.3 opmnctl startall

Syntax: `opmnctl startall [timeout=<seconds>]`

Use this command to start OPMN as well as the OPMN-managed processes for a local Oracle Application Server instance. The `startall` is equivalent to the `start` command and the `startproc` command without arguments. Oracle recommends using the `start` or `startproc` command.

This command operates synchronously and waits for the operation to complete before returning. To set a timeout for the request, specify the `timeout` value in seconds.

Enter the following command for additional detailed information.

```
prompt > opmnctl usage startall  
stopall  
start startall startproc
```

On Microsoft Windows, you can also perform an `opmnctl startall` by starting the `Oracle<OracleHomename>ProcessManager` service in the Windows services control panel. The `Oracle<OracleHomename>ProcessManager` starts automatically when you start or reboot your computer.

2.8.1.4 opmnctl stopall and opmnctl shutdown

Syntax: `opmnctl stopall`, `opmnctl shutdown`

Use either of these commands to shut down the OPMN server as well as the OPMN-managed processes for the local Oracle Application Server instance. This request operates synchronously; it waits for the operation to complete before returning.

The `opmnctl stopall` or the `opmnctl shutdown` command should only be executed prior to shutting down OPMN and your computer.

Shutting down the OPMN server is not necessary during normal operation. Shutting down the OPMN server prevents remote commands to OPMN from executing on the Oracle Application Server instance until OPMN is restarted.

On Microsoft Windows, you can also perform an `opmnctl shutdown` by stopping the `Oracle<OracleHomename>ProcessManager` service in the Windows services control panel.

Use the `opmnctl stopproc` command if you want to stop all OPMN managed processes.

Use the `opmnctl reload` if you want OPMN to reread its configuration.

Enter one of the following commands to obtain additional information.

```
prompt > opmnctl usage stopall
```

or

```
prompt > opmnctl usage shutdown
```

2.8.1.5 opmnctl reload

Syntax: `opmnctl [scope] reload`

Use this command to trigger the OPMN to re-read its configuration files in the requested scope. This command restarts the OPMN server without restarting any Oracle Application Server processes managed by OPMN. The OPMN server for the Oracle Application Server instance must be up and running.

Note: On Microsoft Windows, you can highlight the Oracle<OracleHomename>ProcessManager in the services control panel and select **Restart**. The restart of the service is not equivalent to an `opmnctl reload`, however. This action is equivalent to an `opmnctl shutdown` followed by an `opmnctl startall`. It is a much slower operation than `opmnctl reload` because it restarts OPMN and all the processes managed by OPMN.

Enter the following command for additional detailed information.

```
prompt > opmnctl usage reload
```

See Also: [Section 2.8.3, "Command Definitions"](#)

2.8.2 Process Control Commands

The `opmnctl` process control commands enable you to start, stop, or restart single or multiple Oracle Application Server components. You can control an Oracle Application Server component at the `<ias-component>`, `<process-set>`, or `<process-type>` level.

This section describes the process control commands available with `opmnctl`. It includes the following process control commands:

- [opmnctl startproc](#), [opmnctl restartproc](#) and [opmnctl stopproc](#)

Output is not generated for the successful execution of an `opmnctl` process control command. Refer to [Chapter 17, "OPMN Troubleshooting"](#) if you receive any error messages during `opmnctl` command execution.

2.8.2.1 opmnctl startproc, opmnctl restartproc and opmnctl stopproc

Syntax: `opmnctl [<scope>] startproc [<attr>=<value>...]`

`opmnctl [<scope>] restartproc [<attr>=<value>...]`

`opmnctl [<scope>] stopproc [<attr>=<value>...]`

Use these commands to start, restart, or stop OPMN-managed processes in the requested scope. The OPMN server for the Oracle Application Server instance must be up and running.

The following attributes and values can be used with the `startproc`, `stopproc`, and `restartproc` commands:

- `ias-component`, `process-type`, and `process-set`: The values for these attributes should be the same as the `id` value specified in the `opmn.xml` file. If no attribute is supplied, the command is applied to all OPMN-managed processes.
- `mode`: The `mode` attribute value can be either `sync` or `async`; the default value is `sync`. The `sync` value for mode causes the `opmnctl` command to operate synchronously and wait for the command to be executed completely before a return prompt is displayed. The `timeout` element can only be specified when the value of `mode` is `sync`. The value is specified in number of seconds. After the specified timeout expires, the operation is aborted for `startproc` but not for `restartproc` or `stopproc`. The `opmnctl` command prompt returns, the OPMN server continues to perform the `opmnctl restartproc` or `stopproc` command request until the operation is finished.

The `async` value for `mode` causes the return prompt to be displayed immediately, while the OPMN server continues to perform the `opmnctl` command request until the operation is finished.

- `uniqueid`: This value is assigned by OPMN after starting up. You can use this value when you execute the `restartproc` and `stopproc` commands. You can obtain this value by entering the following command and obtaining the unique number for the Oracle Application Server component in the `uid` column of the generated output.

```
prompt > opmnctl status -l
```

Attribute names other than the those listed may be specified for some types of Oracle Application Server processes managed by OPMN. Unique attribute name should be specific to each type of Oracle Application Server process.

Using the `opmnctl startproc`, `restartproc`, or `stopproc` commands with a specified scope and attributes enables control of specific processes in your enterprise. You can execute the `opmnctl startproc`, `restartproc`, or `stopproc` commands at the `<ias-component>`, `<process-type>` and the `<process-set>` level.

For example, the following command starts OracleAS Wireless at the `<process-set>` level.

```
prompt > opmnctl startproc ias-component=wireless process-type=alert_server  
process-set=alert_instance_1
```

The following command restarts OC4J at the `<process-type>` level.

```
prompt > opmnctl restartproc ias-component=OC4J process-type=home
```

The following command stops Oracle HTTP Server at the `<ias-component>` level.

```
prompt > opmnctl stopproc ias-component=HTTP_Server
```

Enter one of the following commands to obtain additional information.

```
prompt > opmnctl usage startproc
```

or

```
prompt > opmnctl usage restartproc
```

or

```
prompt > opmnctl usage stopproc
```

See Also: ■ [Section 2.8.3, "Command Definitions"](#)

- [Chapter 3, "Using OPMN"](#)

2.8.3 Command Definitions

opmnctl features command definitions that enable you to further define the action you would like to execute with OPMN.

This section describes the command definitions available with the opmnctl command. It includes the following sections:

- [Scope](#)
- [Attributes](#)
- [Verbose](#)
- [Options](#)

2.8.3.1 Scope

Syntax: @instance[:instname[:instname...]]

```
@cluster[:clusname]
```

```
@farm
```

The scope option specifies which Oracle Application Server instances the opmnctl command applies to. You can use the scope option for opmnctl commands for single or multiple Oracle Application Server instances, clusters and farms.

- **@instance:** If you do not specify a name after @instance option, the opmnctl command is applied to the local Oracle Application Server instance; local refers to the Oracle Application Server instance or cluster containing the OPMN server handling the request. The default is the local Oracle Application Server instance. If the @instance option is followed by Oracle Application Server instance names, the request will be routed to Oracle Application Server instances. To apply the command to one or more Oracle Application Server instances, specify @instance[:instname[:instname...]].
- **@cluster:** If you do not specify a name after @cluster option, the opmnctl command is applied to the local Oracle Application Server cluster. If @cluster is followed by a set of 1 or more cluster names, the request will be routed to the all Oracle Application Server instances contained in the specified Oracle Application Server clusters. To apply the command to all Oracle Application Server instances within one or more Oracle Application Server clusters, specify @cluster[:clusname[:clusname...]].

- **@farm:** To apply the command query to all Oracle Application Server instances in the Oracle Application Server farm.

For example, the following command starts OC4J on Oracle Application Server instance named "myInst2.foo.com":.

```
prompt > opmnctl @instance:myInst2.foo.com startproc ias-component=HTTP_Server
```

See Also: [Chapter 3, "Using OPMN"](#)

2.8.3.2 Attributes

syntax: <attribute>=<value>

The opmnctl attributes enable you to apply process control operations to specific Oracle Application Server components. If no attributes are specified, all OPMN-managed processes are started, stopped, or restarted.

For example, the following command starts all Oracle Application Server processes configured for OracleAS Wireless.

```
prompt > opmnctl startproc ias-component=wireless
```

Refer to [Chapter 3, "Using OPMN"](#) for additional opmnctl command examples.

Table 2–2 lists the attribute names and values that OPMN can use this command:

Table 2–2 opmnctl Attribute Names and Values

Attribute Name	Attribute Values
ias-component	Value should be the same as the value for the <code>id</code> attribute for the <code><ias-component></code> element in the <code>opmn.xml</code> file.
process-type	Value should be the same as the value for the <code>id</code> attribute for the <code><process-type></code> element in the <code>opmn.xml</code> file.
process-set	Value should be the same as the value for the <code>id</code> attribute for the <code><process-set></code> element in the <code>opmn.xml</code> file.
mode	Value can either be “sync” or “async”. The default value is “sync”, meaning that this request operates synchronously, and waits for the operation to complete before returning. “async” indicates that the request returns immediately, while OPMN continues to perform the request until the operation finishes.
timeout	This can only be specified in “sync” mode. The value is in seconds. After this timeout expires, OPMN does not continue to perform the request for <code>startproc</code> operations. The request does continue for <code>restartproc</code> and <code>stopproc</code> operations.

See Also: [Chapter 3, "Using OPMN"](#)

2.8.3.3 Verbose

Syntax: `opmnctl verbose command`

The `opmnctl verbose` option enables you to obtain detailed information about the command you are executing.

For example, the following command outputs the information shown in [Example 2–3](#).

```
> opmnctl verbose startproc ias-component=HTTP_Server
```

Example 2–3 opmnctl verbose output

```
HTTP/1.1 200 OK
Content-Length: 0
Content-Type: text/html
Response: Ping succeeded.

opmnctl: starting opmn managed processes...
HTTP/1.1 200 OK
Content-Length: 571
Content-Type: text/html
Response: 1 of 1 processes started.

<response>
<opmn id="jerichar-sun.us.oracle.com:6200" http-status="200" http-response="1 of 1
processes started.">
  <ias-instance id="M140801.jerichar-sun.us.oracle.com">
    <ias-component id="HTTP_Server">
      <process-type id="HTTP_Server">
        <process-set id="HTTP_Server">
          <process id="1954086921" pid="9355" status="Alive" index="1"
log="/home/demoas/M140801/opmn/logs/HTTP_Server~1"
operation="request" result="success">
        </process>
      </process-set>
    </ias-component>
  </ias-instance>
</opmn>
```

```
</process-set>
</process-type>
</ias-component>
</ias-instance>
</opmn>
</response>
```

2.8.3.4 Options

The following are the options you can specify for the <options> parameter:

- **-1:** Use this option to obtain the uniqueid (uid) value.

For example, the following command outputs the information shown in [Example 2–4](#).

```
prompt > opmnctl status -1
```

Example 2–4 opmnctl status -1 output

ias-component	process-type	pid	status	uid	memused	uptime	ports
-OC4J	home	5611	Alive	632225812	105008	17:55:58	jms:3701,rmi:3201,ajp:3000

Wrong: cannot be used for "starting a single" process. Only for stopping and restarting.

The uid information allows you to stop or restart an individual Oracle Application Server process.

For example, the following command stops the home process-type.

```
prompt > opmnctl stopproc uniqueid=632225812
```

- **-fsep <string>:** Use this option to assign a field separator value for your opmnctl status output. The default value is **l**.
- **-rsep <string>:** Use this option to assign a record separator value for your opmnctl status output. The default value is **\n**.
- **-noheaders:** Use this option if you do not want a header displayed after you run the opmnctl status command.
- **-fmt <fmtlist>:** This is a single string containing one or more statistic identifiers connected together where each identifier has the following format: <statname>[<width>{<justification>}]. The default value is:
%cmp18%prt18%pid5R%sta8.

[Table 2–3](#) lists the format string syntax for the <fmtlist> option:

Table 2–3 Format String Syntax

Format String Syntax	Description
<statname>	<p>This must be one of the following:</p> <ul style="list-style-type: none"> ▪ clu: Oracle Application Server cluster name ▪ ins: Oracle Application Server instance name ▪ cmp: Oracle Application Server component ID ▪ prt: process-type ID ▪ prs: process-set ID ▪ idx: index of process in process-set ▪ pid: OS process ID ▪ uid: OPMN uniqueid ▪ typ: name for this kind of process ▪ sta: process status ▪ stm: start time (ms) ▪ utm: up time (ms) ▪ cpu: cpu time (ms) ▪ mem: memory used (in KB) ▪ por: port list
<width>	<p>Specifies the size for the field. Output shorter than this value receives padding according to the specified <justification>. Output longer than this value is truncated, and terminated with '~'.</p> <p>Default: width of each datum.</p>
<justification>	<p>Specifies the justification for the field. This enables you to justify output when it is less than the width. It is L, R, or C (left, right, or center justification).</p> <p>Default: L</p>

For example, the following command displays the output shown in [Example 2–5](#).

```
prompt> opmnctl status -noheaders -fsep @ -fmt %cmp%prt%pid%sta
```

Example 2–5 opmnctl status -noheaders output

```
OC4J@home@N/A@Down
dcm-daemon@dcm-daemon@13875@Alive
LogLoader@logloaderd@9800@Alive
HTTP_Server@HTTP_Server@13926@Alive
```

See Also: [Section 2.8.3, "Command Definitions"](#)

Enter the following command for additional detailed information.

```
prompt > opmnctl usage status
```

2.8.4 Status Commands

The opmnctl status commands enable you to determine the status of OPMN-managed processes.

This section describes the command options available with the opmnctl command. It includes the following sections:

- [opmnctl status](#)
- [opmnctl ping](#)

See Also:

- [Section 2.8.3.1, "Scope"](#)
- [Section 2.8.3.4, "Options"](#)

2.8.4.1 opmnctl status

Syntax: `opmnctl [<scope>] status [<options>]`

The status command enables you to obtain information on the Oracle Application Server processes managed by OPMN.

The output is a text table. Each row in the table represents one Oracle Application Server process.

You can customize the status command in the following ways:

- Supply a scope to obtain status of processes running on other Oracle Application Server instances
- Change the information displayed about each Oracle Application Server process
- Remove the table headers from the output
- Change the field separator
- Change the record separator
- Change the width of individual columns
- Change the justification of the data in an individual column

Enter the `opmnctl usage status` command to obtain full details on how to use the status command.

Example 2–6 shows the output after entering the `opmnctl status` command for the AppSrv1 instance on host comp1 for the domain `yourcompany.com`:

Example 2–6 opmnctl Status Output

```
prompt > opmnctl status
```

Processes in Instance: AppSrv1.comp1.yourcompany.com			
ias-component	process-type	pid	status
WebCache	WebCacheAdmin	29121	Alive
WebCache	WebCache	29120	Alive
OC4J	OC4J_Demos	N/A	Down
OC4J	home	29268	Init
dcm-daemon	dcm-daemon	29113	Alive
LogLoader	logloader	N/A	Down
HTTP_Server	HTTP_Server	29099	Alive

You can use the `opmnctl status` command with `<scope>` to obtain additional detailed information. For example, the following command gives you the status of every process of every component of every OracleAS Instance in an entire farm.:

```
opmnctl @farm status
```

See Also: [Section 2.8.3.1, "Scope"](#)

2.8.4.2 opmnctl ping

Syntax: `opmnctl ping [<max_retry>]`

The `opmnctl ping` command enables you to contact the local OPMN server to verify operation. `<max_retry>` specifies the maximum number of retry times. If `<max_retry>` is specified, the local OPMN is pinged every one second, until the command execution succeeds or `max_retry` is reached.

For example, the following command,

```
prompt > opmnctl ping 10
```

designates pinging of OPMN 10 times until the ping command succeeds

2.8.5 Help Commands

The `opmnctl help` commands enable you to obtain additional information regarding OPMN.

This section describes the help command options available with the `opmnctl` command. It includes the following sections:

- [opmnctl help](#)
- [opmnctl usage](#)
- [opmnctl validate](#)

2.8.5.1 opmnctl help

Syntax: `opmnctl help`

Use this command to print a short syntax description of `opmnctl` commands.

[Example 2–7](#) shows the output from the opmnctl help command.

Example 2–7 opmnctl help Output

prompt > opmnctl help

```
usage: /ORACLE_HOME/bin/opmnctl [verbose] [<scope>] <command> [<options>]
```

verbose: print detailed execution message if available

Permitted <scope>/<command>/<options> combinations are:

scope	command	options	
	start		- Start opmn
	startall		- Start opmn & all managed processes
	stopall		- Stop opmn & all managed processes
	shutdown		- Shutdown opmn & all managed processes
[<scope>]	startproc	[<attr>=<val>...]	- Start opmn managed processes
[<scope>]	restartproc	[<attr>=<val>...]	- Restart opmn managed processes
[<scope>]	stopproc	[<attr>=<val>...]	- Stop opmn managed processes
[<scope>]	reload		- Trigger opmn to reread opmn.xml
[<scope>]	status	[<options>]	- Get managed process status
	ping	[<max_retry>]	- Ping local opmn
	validate	[<filename>]	- Validate the given xml file
	help		- Print brief usage description
	usage	[<command>]	- Print detailed usage description

2.8.5.2 opmnctl usage

Syntax: opmnctl usage [<command>]

The usage command displays help for all opmnctl commands, or only for the specified command.

The command can be one or more of the following:

- start
- startall
- startproc
- stopall
- stopproc
- restartproc
- reload
- shutdown
- ping
- status
- help

For example, enter the following command to receive the output shown in [Example 2-8](#).

```
prompt > opmnctl usage stopall
```

Example 2-8 opmnctl usage stopall output

opmnctl stopall

Stop opmn daemon and opmn managed processes for local ias instance.

This request first tries to stop all opmn managed processes gracefully. Processes which will not stop gracefully will be forcefully shutdown. After stopping all managed processes, the opmn daemon will shutdown itself.

This request should only be performed when it is necessary to stop the opmn daemon. Once started, the opmn daemon should remain up until it is necessary to reboot the computer or some other rare administrative event occurs.

To stop all opmn managed processes without stopping the opmn daemon, consider using the stopproc command without any arguments.

To restart the opmn daemon without restarting any managed processes, consider using the the reload command. The reload command is the appropriate command to use when the only goal is to restart the opmn daemon with a new configuration.

This request operates synchronously and will wait for the operation to complete before returning.

2.8.5.3 opmnctl validate

Syntax: opmnctl validate [<filename>]

The opmnctl validate command validates the XML syntax of the opmn.xml file. The default *ORACLE_HOME*/opmn/conf/opmn.xml is validated if the filename parameter is not specified. The <filename> can be specified by either the relative or absolute path.

Only one file can be validated at a time. No output is generated for successful opmn.xml file validation.

2.9 OPMN Log Files

The log files generated by OPMN provide important information that can help you identify and diagnose performance and configuration issues. The Application Server Control Console makes reviewing these log files easier by helping you locate and view Oracle Application Server component log files:

See Also:

- [Section 17.1.1, "OPMN log Files"](#)
- [Oracle Application Server 10g Administrator's Guide](#)

2.10 Security

The OPMN local listener port used by ONS clients and PM administrative processes do not use Secure Socket Layer (SSL) encryption for security, but rely on two other mechanisms to ensure authorized access to the OPMN server:

- OPMN binds the local listener port to the local host. Users on the local system can connect to this port and issue OPMN process control requests. Information requests are allowed on the OPMN request port, which is bound to the system IP. The request port does not have SSL encryption.
- When the OPMN server process first starts up and successfully binds to the local port, it creates a string of printable ASCII characters which it uses as a key for local connections. All connection attempts on the local port must include this key or the connection is closed by the OPMN server. The ASCII character string is written into the `ORACLE_HOME/opmn/conf/.formfactor` file. Processes that cannot access the `.formfactor` file are not permitted to interact with the OPMN server.
- OC4J also uses the OPMN `.formfactor` file for similar request validation.

For security reasons, the OPMN server logs any attempts to connect to its local port with an invalid form factor key (a key that does not match the value written by this OPMN process into the `.formfactor` file).

In addition to attempted security violations, there are three common user errors that can cause this error to occur.

1. The user attempts to run the OPMN client manually with the wrong user identification. Only the application server user can read the value from the `.formfactor` file, and so requests or processes run as the wrong user will not be able to provide the correct key to the OPMN server.
2. The user is attempting to run an OPMN client from the wrong `ORACLE_HOME`. It is possible to have multiple `ORACLE_HOME` instances set up on the same system. If the other `ORACLE_HOME` instances have OPMN configured to use the same local port then the Oracle Application Server process request from the wrong `ORACLE_HOME` will read the wrong `.formfactor` file.
3. The user has manually changed the local port configuration in the `opmn.xml` file and started a new OPMN server without first stopping the previous OPMN server. The new OPMN server will run, bind to the new port, and overwrite the `.formfactor` file. The previous OPMN server is now unreachable via the local port, and can only be shutdown via remote OPMN requests (if SSL and authentication are configured) or by manually stopping the previous OPMN server.

2.10.1 Remote Security

OPMN supports remote requests to other OPMN servers in the same farm, but for security reasons all process control requests (start, restart and stop) are only enabled if SSL is enabled in the `opmn.xml` file and a wallet file is configured. If neither SSL nor a wallet file are configured, OPMN will reject any remote process control request with HTTP code 403.

The remote port used for remote administration must be SSL-enabled. The remote port should only be used for communication between multiple OPMN servers. Oracle Application Server components and Application Server Control Console transmit through the local port which is inaccessible to remote administration. All access control and authentication is controlled by going through Application Server Control Console.

See Also: *Oracle Application Server 10g Security Guide*

2.11 iHAT

The Oracle Application Server Hi-Av Tool (iHAT) provides a real time, graphical interface view of your enterprise. iHAT displays all Oracle Application Server processes managed by one or more OPMN servers including useful performance metrics about each process. The snapshot of the system is updated continuously at a configurable interval.

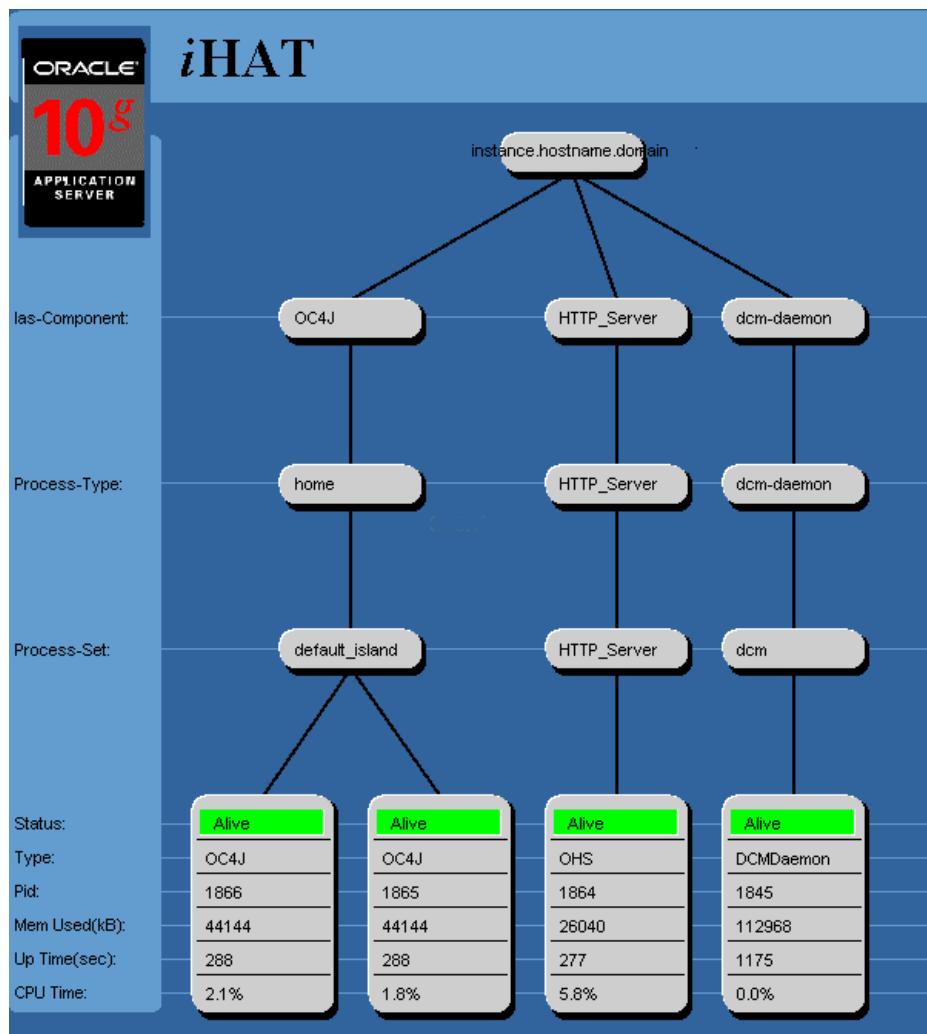
iHAT shows your Oracle Application Server instance with Oracle Application Server components. The `ias-component`, `process-set`, and `process-type` levels are displayed for each Oracle Application Server component. The child process display of the `process-type` shows the status, `process-type`, process identification (pid), memory usage, uptime, and CPU usage.

[Figure 2-2](#) shows an example of the iHAT display for an enterprise with two Oracle Application Server instances.

To download iHAT, visit the Oracle Technology Network (OTN):

<http://otn.oracle.com/membership>

Figure 2–2 Example iHAT Display



3

Using OPMN

This chapter provides command-line examples on how to use OPMN for Oracle Application Server. It features the following topics:

- Starting OPMN
- Starting and Stopping OPMN Managed Processes for a Local Oracle Application Server Instance
- Starting and Stopping all OPMN Managed Processes for a Remote Oracle Application Server Instance
- Starting and Stopping an Oracle Application Server Component in a Local Oracle Application Server Instance
- Starting and Stopping an Oracle Application Server Process Type in a Local Oracle Application Server Instance
- Starting and Stopping a Multi-Oracle Application Server Instance Environment
- Starting a Component on an Oracle Application Server Farm

3.1 Starting OPMN

OPMN does not depend on any other Oracle Application Server component being up and running before it can be started and used. The OPMN server should be started as soon as possible after turning on the host.

Use the following command to start OPMN without starting other Oracle Application Server components:

```
prompt > opmnctl start
```

3.2 Starting and Stopping OPMN Managed Processes for a Local Oracle Application Server Instance

Use the following command to **start** OPMN managed processes for a local Oracle Application Server instance:

```
prompt > opmnctl startproc
```

Use the following command to **stop** OPMN managed processes for a local Oracle Application Server instance:

```
prompt > opmnctl stopproc
```

Note: Without arguments the `opmnctl startproc` and `opmnctl stopproc` commands start and stop all OPMN managed processes.

3.3 Starting and Stopping all OPMN Managed Processes for a Remote Oracle Application Server Instance

Use the following command to **start** OPMN and OPMN managed processes for a remote Oracle Application Server instance:

```
prompt > opmnctl @instance:oracleas2.foo.com startproc
```

Use the following command to **stop** OPMN and OPMN managed processes for a remote Oracle Application Server instance:

```
prompt > opmnctl @instance:oracleas2.foo.com stopproc
```

3.4 Starting and Stopping an Oracle Application Server Component in a Local Oracle Application Server Instance

Use the following command to **start** Oracle Internet Directory in a local Oracle Application Server instance:

```
prompt > opmnctl startproc ias-component=OID
```

Use the following command to **stop** Oracle Internet Directory in a local Oracle Application Server instance:

```
prompt > opmnctl stopproc ias-component=OID
```

3.5 Starting and Stopping an Oracle Application Server Process Type in a Local Oracle Application Server Instance

Use the following command to **start** the performance_server process-type in a local Oracle Application Server instance:

```
prompt > opmnctl startproc ias-component=wireless process-type=performance_server
```

Use the following command to **stop** the performance_server process-type in a local Oracle Application Server instance:

```
prompt > opmnctl stopproc ias-component=wireless process-type=performance_server
```

3.6 Starting and Stopping a Multi-Oracle Application Server Instance Environment

Use the following command to **start** a multi-Oracle Application Server instance environment from local instance oracleas1:

```
prompt > opmnctl @instance:oracleas1.foo.com:oracleas2.bar.com startproc
```

This command starts all processes of all components on all three instances specified with the <scope> argument. Notice that the local instance oracleas1 is specified in the command.

Use the following command to **stop** a multi-Oracle Application Server instance environment from local instance oracleas1:

```
prompt > opmnctl @instance:oracleas1.foo.com:oracleas2.bar.com stopproc
```

This command stops all processes of all components on all three instances specified with the <scope> argument. Notice that the local instance oracleas1 is specified in the command.

Note: You must also indicate your local Oracle Application Server instance when using the @instance <scope> with other Oracle Application Server instances. Commands with a <scope> argument only operates on the instances described by the <scope> argument. The command will only be applied to the local instance if it is described in the <scope> argument.

3.7 Starting a Component on an Oracle Application Server Farm

Use the following command to **start** the same Oracle Application Server component on multiple Oracle Application Server instances:

```
prompt > opmnctl @farm startproc ias-component=HTTP_Server
```

Use the following command to **stop** the same Oracle Application Server component on multiple Oracle Application Server instances:

```
prompt > opmnctl @farm stopproc ias-component=HTTP_Server
```


4

opmn.xml Common Configuration

This chapter provides common configuration examples, and descriptions of elements and attributes for the OPMN `opmn.xml` file.

It contains the following topics:

- [Example of opmn.xml Elements and Attributes](#)
- [opmn.xml Element and Attribute Descriptions](#)

4.1 Example of opmn.xml Elements and Attributes

[Example 4-1](#) shows all possible elements and attributes that may appear in an opmn.xml file that are not specific to any Oracle Application Server component.

Note: OPMN will convert slashes in the path value string to be those of the directory path separator character for the system on which OPMN is running (for UNIX each \ character is converted to /; for Windows each / is converted to \).

OPMN uses the ^ character as an escape character to disable slash conversion. ^/ on a Windows system will yield a / in the string. Specify two ^ characters if you need to specify the ^ character in the resultant string. For example, ^^ yields ^.

Example 4-1 Common Configuration Elements and Attributes

```
<opmn>
  <notification-server>
    <ipaddr local="ip" remote="ip" request="ip"/>
    <port local="port" remote="port" request="port"/>
    <log-file path="path" level="level" rotation-size="kBytes" rotation-hour="HOD"/>
    <ssl enabled="boolean" wallet-file="path" wallet-password="password"/>
  </notification-server>
  <process-manager insecure-remote-requests="boolean">
    <log-file path="path" level="level" rotation-size="kBytes" rotation-hour="HOD"/>
  </process-modules>
    <module path="path" tag="tag-id" status="state" cron="interval">
      <module-data>
        <category id="id">
          <data id="id" value="value" process-conversion="boolean"/>
        </category>
        </module-data>
      <module-id id="module-id"/>
    </module>
  </process-modules>
  <ias-instance id="ias-instance-name" ORACLE_HOME="path">
    <environment>
      <variable id="id" value="value" append="boolean" process-conversion="boolean"/>
    </environment>
    <!-- module-data -->
    <ias-component id="component-id" id-matching="boolean" status="state">
      <!-- environment -->
      <!-- module-data -->
      <dependencies>
        <database db-connect-info="connect" infrastructure-key="key"
          timeout="depend-timeout" cache-timeout="cache-timeout"/>
        <OID address="address" timeout="depend-timeout" cache-timeout="cache-timeout">
          <ssl enabled="boolean" wallet-file="path" wallet-password="password"/>
        </OID>
        <OSO host="hostname" port="port" URI="uri" timeout="depend-timeout"
          cache-timeout="cache -timeout"/>
          <ssl enabled="boolean" wallet-file="path" wallet-password="password"/>
        </OSO>
        <managed-process ias-instance="ias-instance-id" ias-component="ias-component-id"
          process-type ="process-type-id" process-set="process-set-id" autostart="boolean"
          autostop="boolean" timeout="depend-timeout" cache-timeout="cache-timeout"/>
      </dependencies>
      <process-type id="process-type-id" module-id="module-id" status="state">
    </process-type>
  </ias-instance>
</opmn>
```

```
working-dir="path">
    <!--environment -->
    <!-- module-data -->
    <!-- dependencies -->
    <event-scripts>
        <pre-start path="path">
        <pre-stop path="path">
        <post-crash path="path">
    </event-scripts>
    <start timeout="timeout" retry="num"/>
    <stop timeout="timeout"/>
    <restart timeout="timeout" retry="num"/>
    <ping timeout="timeout" retry="num" interval="interval"/>
    <port id="id" range="range"/>
    <process-set id="process-set-id" restart-on-death="boolean" numprocs="num"
status="state" working-dir="path">
        <!-- environment -->
        <!-- module-data -->
        <!-- dependencies -->
        <!-- event script -->
        <!-- start -->
        <!-- stop -->
        <!-- restart -->
        <!-- ping -->
        <!-- port -->
    </process-set>
</process-type>
</ias-component>
</ias-instance>
</process-manager>
</opmn>
```

4.2 opmn.xml Element and Attribute Descriptions

This section describes the elements and attributes in the `opmn.xml` file that are not specific to any Oracle Application Server component. This section also provides attribute descriptions of the elements.

`<opmn>`

Required: true
Default: none
Parents: none
Attributes: none

Top-level element for `opmn.xml`

`<notification-server>`

Required: true
Default: none
Parents: `<opmn>`
Attributes: none

Configures or, contains elements to configure the ONS portion of OPMN.

`<ipaddr>`

Required: true
Default: none
Parents: `<notification-server>`
Attributes: local, remote, request

Specifies host information for ONS listener threads and host port bindings.

`local= "ip or hostname"`

Required: true
Default: none
Valid Values: IP address (in `###.###.###.###` format) or host name to which ONS will bind its local port.

IP address or host name to which ONS will bind its local port. All local OPMN requests are routed through the local port. All local applications connect to ONS through the local port to send and receive notifications.

`remote="ip or hostname"`

Required: false
Default: none
Valid Values: IP address (in `###.###.###.###` format) or host name to which ONS will bind its remote port.

IP address or host name to which ONS will bind its remote port. The remote port is used for ONS to ONS communication. Notifications pass from ONS to ONS via the remote port, and OPMN uses ONS to route remote requests to other OPMNs via the remote port.

request="ip or hostname"

Required: false

Default: none

Valid Values: IP address (in ###.###.###.### format) or host name to which ONS will bind its request port.

IP address or host name to which ONS will bind its request port. This port is only used for Oracle Dynamic Monitoring Service (DMS) dump requests (and the obsolete global-dumps in Oracle9iAS Release 2 (9.0.2/9.0.3) from some OC4J configurations and some monitoring tools. Only dump requests will be processed from this port.

For example:

```
<ipaddr local="ip" remote="ip" request="ip"/>
```

<port>

Required: true

Default: none

Parents: [<notification-server>](#)

Attributes: local, remote, request

Configuration of the port information for ONS listener threads host and port bindings.

local="port"

Required: true

Default: none

Valid Values: A port number.

ONS local port value.

remote="port"

Required: false

Default: none

Valid Values: A port number.

ONS remote port value.

request="port"

Required: false

Default: none

Valid Values: A port number.

ONS request port value.

<log-file>

Required: true

Default: none

Parents: [<notification-server>](#)

Attributes: path, level, rotation-size, rotation-hour

Configuration definitions for the ONS log mechanism.

path="path"

Required: true

Default: none

Valid Values: A file name path for the ONS log file.

The path attribute is a value string. All directories specified in the path must already exist, and OPMN must have read and write permissions for the directory in which the log file resides. \$ORACLE_HOME may be used.

ORACLE_HOME is the root directory in which Oracle software is installed.

OPMN will convert slashes in the path value string to be those of the directory path separator character for the system on which OPMN is running (for UNIX each \ character is converted to /; for Windows each / is converted to \).

OPMN uses the ^ character as an escape character to disable slash conversion. ^/ on a Windows system will yield a / in the string. Specify two ^ characters if you need to specify the ^ character in the resultant string. For example, ^^ yields ^.

level="level"

Required: true

Default: none

Valid Values: An integer value of 0 through 9.

This value specifies the ONS logging level.

0 No logging

1 Fatal errors

2 Errors

3 Warnings

4 Events worth noting

5 Listener and resource management, opmn.xml parse information

6 Connection management, internal notification routing, and worker thread management

7 Notification parsing and processing, subscription addition and removal, and high level send queue checking

8 Received notification content, detailed send queue processing, detailed subscription processing

9 Sent notification content and detailed subscription processing

rotation-size="kBytes"

Required: false

Default: none

Valid Values: An integer.

The maximum size in kilobytes of the log file. When the log file reaches the configured size, the ONS logging mechanism will close the log file, rename it with an integer time stamp, and then create a new log file. This attribute may be used with rotation-hour.

rotation-hour="HOD"

Required: false

Default: none

Valid Values: An integer value of 0 through 23.

At the prescribed hour of the day, the ONS logging mechanism will close the log file, rename it with an integer time stamp, and then create a new log file. This attribute may be used with rotation-size.

<ssl>

Required: false

Default: none

Parents: [<notification-server>](#)

Attributes: enabled, wallet-file, wallet-password

enabled="boolean"

Required: true

Default: none

Valid Values: true or false

If the value is true, enables SSL for ONS.

wallet-file="path"

Required: false

Default: none

Valid Values: A path to an Oracle wallet.

Specify the Oracle wallet to use for authentication on ONS connections. \$ORACLE_HOME may be used.

OPMN will convert slashes in the path value string to be those of the directory path separator character for the system on which OPMN is running (for UNIX each \ character is converted to /; for Windows each / is converted to \).

OPMN uses the ^ character as an escape character to disable slash conversion. ^/ on a Windows system will yield a / in the string. Specify two ^ characters if you need to specify the ^ character in the resultant string. For example, ^^ yields ^.

wallet-password="password"

Required: false

Default: none

Valid Values: A string for the wallet password.

The password string for the specified wallet.

<process-manager>

Required: true

Default: none

Parents: [<opmn>](#)

Attributes: insecure-remote-requests

The process-manager contains the configuration definitions for the PM portion of OPMN. The

insecure-remote-requests="boolean"

Required: false

Default: false

Valid Values: true or false

By default OPMN will only allow start, stop, restart, shutdown and reload requests rerouted from remote OPMNs if ONS SSL is enabled and a wallet file is configured for authentication.

Note: Setting this attribute to true overrides that security check and allows these requests to be issued remotely with no security features configured.

Setting this attribute to true is a major security risk and should only be done for testing purposes with all connected OPMNs behind a well secured fire wall or completely disconnected from any external network.

<log-file>

Required: true

Default: none

Parents: [**<process-manager>**](#)

Attributes: path, level, rotation-size, rotation-hour

Configuration definitions for the PM log mechanism.

path="path"

Required: true

Default: none

Valid Values: A file name path for the PM log file.

All directories specified in the path must already exist, and OPMN must have read and write permissions for the directory in which the log file resides. \$ORACLE_HOME may be used.

OPMN will convert slashes in the path value string to be those of the directory path separator character for the system on which OPMN is running (for UNIX each \ character is converted to /; for Windows each / is converted to \).

OPMN uses the ^ character as an escape character to disable slash conversion. ^/ on a Windows system will yield a / in the string. Specify two ^ characters if you need to specify the ^ character in the resultant string. For example, ^^ yields ^.

level="level"

Required: true

Default: none

Valid Values: An integer value of 0 through 9.

This value specifies the PM logging level.

0 No logging

1 Fatal errors

2 Errors

3 Warnings

4 Events worth noting

5 Internal and module initialization, process monitoring, and dependency check results

6 Process resource management, http request and process job scheduling and execution, and dependency check details

7 Http request parsing and resource management

8 Http request response, and DMS statistics collection and requests

9 Low level thread tracing

`rotation-size="kBytes"`

Required: false

Default: none

Valid Values: An integer.

The maximum size in kilobytes of the log file. When the log file reaches the configured size, the PM logging mechanism will close the log file, rename it with an integer time stamp, and then create a new log file. This attribute may be used with `rotation-hour`.

`rotation-hour="HOD"`

Required: false

Default: none

Valid Values: An integer value of 0 through 23.

At the prescribed hour of the day, the PM logging mechanism will close the log file, rename it with an integer time stamp, and then create a new log file. This attribute may be used with `rotation-size`.

<process-modules>

Required: true

Default: none

Parents: [<process-manager>](#)

Attributes: none

The PM dynamically loads in a library for each specified process module. Each process module is designed to support a specific set of `process-type`, and is only required if those `process-type` are configured.

<module>

Required: true

Default: none

Parents: [<process-modules>](#)

Attributes: path, tag, status, cron

A module is used to provide `process-type` specific support for the PM. Each module is implemented as a shared library which exports a set of standard functions and uses the PM process module API. A module must provide a list of the `process-types` it supports, and only one configured process module may list a specific `process-type`; no two modules can list the same `process-type`.

path="path"

Required: true

Default: none

Valid Values: A path for the module shared library.

The path must specify the shared library file, but if the library file has the standard system suffix (.so for UNIX and .dll for Windows), the suffix may be omitted and OPMN will automatically append it. \$ORACLE_HOME or %ORACLE_HOME% may be used when specifying the path..

OPMN will convert slashes in the path value string to be those of the directory path separator character for the system on which OPMN is running (for UNIX each \ character is converted to /; for Windows each / is converted to \).

OPMN uses the ^ character as an escape character to disable slash conversion. ^/ on a Windows system will yield a / in the string. Specify two ^ characters if you need to specify the ^ character in the resultant string. For example, ^^ yields ^.

tag="tag-id"

Required: false

Default: The value specified by path.

Valid Values: A string uniquely identifying the module.

A module may report its tag value when logging errors to the PM log file or as part of the response to a request. While optional, it is a good idea to set this attribute to a meaningful value to help track any issues with process management.

status="state"

Required: false

Default: enabled

Valid Values: critical, enabled, or disabled

A module may be enabled, in which case PM loads in its shared library when it starts and calls the module's initialization functions, or disabled in which case the module entry is completely ignored. If the module process-types are configured in opmn.xml they must also be disabled. The critical state is the same as enabled, except that OPMN will terminate with a fatal error code if the module initialization fails.

cron="interval"

Required: false

Default: none

Valid Values: An integer.

Specify the interval in seconds between calls to the module's cron callback function. Configuring a cron interval for a module that does not support the cron callback is not allowed. Unless you have designed the module, you should neither add nor alter this attribute.

<module-data>

Required: false

Default: none

Parents: [<module>](#) , [<ias-instance>](#) , [<ias-component>](#) , [<dependencies>](#) , [<process-set>](#)

Attributes: none

The module-data blocks are used to define module specific name-value pairs that are meaningful only to a specific module. Each module-data block is organized into categories, which contain the name-value data pairs.

The module-data blocks can be defined for multiple elements within opmn.xml, and OPMN will create an aggregate module-data block at the process-set level that contains all values defined at or above it. If multiple definitions exist in this hierarchy with the same category id and data id, the value defined at the lowest level is used.

Table 4–1 illustrates the module-data defined at each level in the hierarchy (with the highest level displayed at the top) and the resultant union at the process-set level of all of the module-data definitions:

Table 4–1 module-data Hierarchy

Module	Definition
ias-instance	<pre><category id="CatA"> <data id= "DataAA" value="aaaa"/> </category></pre>
ias-component	<pre><category id="CatA"> <data id= "DataAB" value="abab"/> </category> <category id="CatB"> <data id= "DataBA" value="baba"/> </category></pre>
module	<pre><category id="CatA"> <data id= "DataAC" value="acac"/> </category></pre>
process-type	<pre><category id="CatA"> <data id= "DataAA" value="XXXX"/> </category></pre>
process-set	<pre><category id="CatB"> <data id= "DataBB" value="bbbb"/> </category></pre>
RESULT	<pre><category id="CatA"> <data id= "DataAA" value="XXXX"/> <data id= "DataAB" value="abab"/> <data id= "DataAC" value="acac"/> </category> <category id="CatB"> <data id= "DataBA" value="baba"/> <data id= "DataBB" value="bbbb"/> </category></pre>

<category>

Required: true

Default: none

Parents: [<module-data>](#)

Attributes: id

The category is an organizational level within a module-data block.

id="id"

Required: true

Default: none

Valid Values: A string.

This string identifies a data category. Each category id within a single module-data block must be unique, but multiple module-data blocks may contain the same data category ids, in which case the categories are considered to be related.

<data>

Required: true

Default: none

Parents: [<category>](#)

Attributes: id, value, process-conversion

A data name value definition within a module-data category.

id="id"

Required: true

Default: none

Valid Values: A string.

This string identifies a data element. Each data id within a single category must be unique, but multiple categories may contain the same data identifications. Data elements with the same identification as others, defined in different categories with the same identification are related.

value="value"

Required: true

Default: none

Valid Values: A string.

The value string associated with the data element id. Any environment variable defined anywhere within the scope of the process-set (any level at or above the process-set) in which the data value is defined (again, any level at or above the process-set) referenced within the value string as \$variable will be expanded to the variable value.

process-conversion="boolean"

Required: false

Default: true

Valid Values: true or false

By default OPMN converts slashes in the data value string to be those of the directory path separator character for the system on which OPMN is running (on UNIX each '\' character is converted to '/' and on Windows each '/' is converted to '\'). Set this attribute to false to disable conversion.

<module-id>

Required: true
 Default: none
 Parents: [<module>](#)
 Attributes: id

The module-id name defines the type of process and associates the configuration with a process module.

This identifier is used by each process-type to specify which module supports it. A module may be configured with multiple module-ids.

id="module-id"

Required: true
 Default: none
 Valid Values: A string.

<ias-instance>

Required: true
 Default: none
 Parents: [<process-manager>](#)
 Attributes: id, ORACLE_HOME

The configuration definitions for an Oracle Application Server instance. Only one ias-instance is supported per OPMN.

id="ias-instance-name"

Required: true
 Default: none
 Valid Values: A string.

This string should match the DCM configured ias-instance name in dcm.conf. OPMN itself ignores the value of this string and always uses the DCM configured value.

ORACLE_HOME="path"

Required: true
 Default: none
 Valid Values: A path string.

This path must be the \$ORACLE_HOME equivalent for this Oracle Application Server instance.

OPMN will convert slashes in the path value string to be those of the directory path separator character for the system on which OPMN is running (for UNIX each \ character is converted to /; for Windows each / is converted to \).

OPMN uses the ^ character as an escape character to disable slash conversion. ^/ on a Windows system will yield a / in the string. Specify two ^ characters if you need to specify the ^ character in the resultant string. For example, ^^ yields ^.

<environment>

Required: true

Default: Refer to the following paragraph.

Parents: [<ias-instance>](#), [<ias-component>](#), [<dependencies>](#), [<process-set>](#)

Attributes: none

Like module-data blocks, environment blocks can be defined for multiple elements within the opmn.xml file, and OPMN will create an aggregate environment block at the process-set level that contains all values defined at, or above it. If multiple definitions exist in this hierarchy with the same id, the value defined at the lowest level is used.

Note: OPMN sets the following default environment variables at the ias-instance level, with the values extracted either from the ias-instance configuration or from the OPMN run time environment:

UNIX: ORACLE_HOME, ORACLE_NLS, OPMN_ENV_LC_ALL, OPMN_ENV_LANG, OPMN_ENV_NLS_LANG, SHELL

Windows: COMSPEC, SYSTEM_DRIVE, and SYSTEM_ROOT

<variable>

Required: true

Default: none

Parents: [<environment>](#)

Attributes: id, value, append, process-conversion

The environment variable name and value are defined with this element.

id="name"

Required: true

Default: none

Valid Values: A string.

The environment variable name. An environment id may be duplicated within an environment block, with the last defined value taking priority over earlier definitions. The same environment id may be defined within environment blocks for different elements, and the value defined at the lowest level will take priority over values defined at higher levels.

value="value"

Required: true

Default: none

Valid Values: A string.

The environment value. Environment variables referenced within the value string as \$variable will be expanded to the variable value. The same environment variable may reference itself to use a definition defined at a higher level, or earlier within this same environment block.

You may use the UNIX shell syntax for referencing an environment variable, \$variable or \${variable}, or the Windows format %variable%. Referenced variables that have not been defined remain in place as referenced, and so value="_notdefined_" would remain unchanged if notdefined was not defined.

For example, the following environment block yields a value for accumulate of "foobar".

```
<environment>
  <variable id="accumulate" value="foo">
  <variable id="accumulate" value="${accumulate}bar">
</environment>
```

append="boolean"

Required: false

Default: false

Valid Values: true or false

You can force OPMN to append the new environment variable value to the previously defined value, with the system library delimiter placed in between the two values (':' for UNIX and ';' for Windows) by specifying a value of true for this attribute. This is useful when assembling a value for a variable such as CLASSPATH.

For example, the following environment block yields a value for CLASSPATH of "/foo:/bar" on a UNIX system.

```
<environment>
  <variable id="CLASSPATH" value="/foo">
  <variable id="CLASSPATH" value="/bar" append="true">
</environment>
```

process-conversion="boolean"

Required: false

Default: true

Valid Values: true or false

OPMN converts slashes in the environment value string to be those of the directory path separator character for the system on which OPMN is running (for UNIX each \ character is converted to /; for Windows each / is converted to \).

If the process conversion is true, OPMN uses the ^ character as an escape character to disable slash conversion. ^/ on a Windows system will yield a / in the string. Specify two ^ characters if you need to specify the ^ character in the resultant string. For example, ^^ yields ^.

<ias-component>

Required: true

Default: none

Parents:<[ias-instance](#)>

Attributes: id, id-matching, status

An ias-component is a logical grouping of process-type for administrative purposes.

id="component-id"

Required: true

Default: none

Valid Values: a string

The id attribute uniquely identifies this ias-component within the ias-instance.

id-matching="boolean"

Required: false

Default: false

Valid Values: true or false

By default OPMN requests that do not specify ias-components match all configured ias-components, unless the id-matching attribute for a component is set to true, in which case the request must explicitly include the ias-component id in order to affect the ias-component or any process-type or process-set configured for that ias-component.

status="state"

Required: false

Default: enabled

Valid Values: enabled or disabled

An ias-component may be enabled, in which case OPMN parses all of its configured attributes and elements and allows requests to operate upon it, or disabled, in which case the ias-component entry is completely ignored.

<dependencies>

Required: false

Default: none

Parents: [<ias-component>](#), [<dependencies>](#), [<process-set>](#)

Attributes: none

OPMN uses dependencies to determine if a process should be started or not. Like module-data, environment blocks can be defined for multiple elements within opmn.xml. OPMN will create an aggregate dependency list at the process-set level that contains all dependencies defined at or above it. If duplicate dependencies are defined at different levels, then duplicate checks on the dependency will be made before starting a process.

There are two primary types of dependencies: external and internal. External dependencies are for those resources not managed by OPMN. For example: OracleAS Metadata Repository and Application Server Control Console.

An external program is executed by OPMN to perform the check on the resource. Internal dependencies are for OPMN-managed processes (unit), which may include processes managed on a remote OPMN.

OPMN maintains a cache of dependency states which contains the last known state of each dependency, and the time it was last checked. A cache-timeout parameter for each dependency allows users to specify how long to use its state in the cache, or if it should be used at all. Similarly, a general timeout parameter for each dependency will determine how long OPMN will wait for a status update from that dependency before aborting the dependency check and the process start.

OPMN checks dependencies in the order in which they are declared. The traversal of this list of dependencies concludes either with the full sequence of successful checks, the dependency is available, or the first check failure, the dependency is not available, or the dependency check timed out.

<database>

Required: false

Default: none

Parents: [<dependencies>](#)

Attributes: db-connect-info, infrastructure-key, timeout, cache-timeout

Specifies the database to check: either db-connect-info or infrastructure-key is used to identify the database.

db-connect-info="connect"

Required: true if infrastructure-key is not specified.

Default: none

Valid Values: A string

The string required to connect to the database.

infrastructure-key="key"

Required: true if db-connect is not specified.

Default: none

Valid Values: A string

The infrastructure key required to identify the database.

timeout="depend-timeout"

Required: false

Default: 1200

Valid Values: An integer

The timeout attribute specifies in seconds how long OPMN will wait for a dependency check to complete. If the check takes longer than the configured timeout, then OPMN will consider the check to have failed.

cache-timeout="cache-timeout"

Required: false

Default: 600

Valid Values: An integer

The cache-timeout attribute specifies how long in seconds OPMN will use the current "up" status for this dependency's entry in the cache. If the last successful dependency check was within the prescribed number of seconds from the current check, then the dependency check is instantly flagged as successful, otherwise another dependency check will be performed. Note that the cache-timeout is only for the last successful check of the dependency, and if the previous check failed, another access of the dependency will be performed for this check. A value of 0 indicates OPMN will always perform the check.

<OID>

Required: false

Default: none

Parents: [<dependencies>](#)

Attributes: address, infrastructure, timeout, cache-timeout

Specifies the Oracle Internet Directory (OID) service to check either an address string for a specific Oracle Internet Directory, or that the OracleAS Infrastructure flag is set to true to use the default infrastructure Oracle Internet Directory.

address="address"

Required: true

Default: none

Valid Values: A string

The address string required to connect to Oracle Internet Directory.

infrastructure="boolean"

Required: true if address is not set.

Default: none

Valid Values: true or false

Use the default infrastructure Oracle Internet Directory for this Oracle Application Server instance.

timeout="depend-timeout"

Required: false

Default: 1200

Valid Values: An integer

The timeout attribute specifies in seconds how long OPMN will wait for a dependency check to complete. If the check takes longer than the configured timeout, then OPMN considers the check to have failed.

cache-timeout="cache-timeout"

Required: false

Default: 600

Valid Values: An integer

The cache-timeout attribute specifies how long in seconds OPMN will use the current "up" status for the dependency entry in the cache. If the last successful dependency check was within the prescribed number of seconds from the current check, then the dependency check is flagged as successful. Otherwise, OPMN performs another dependency check. The cache-timeout is only for the last successful check of the dependency. If the previous check failed, OPMN performs another access of the dependency check. A value of 0 indicates OPMN will always perform the check.

<ssl>

Required: false

Default: none

Parents: [<OID>](#)

Attributes: enabled, wallet-file, wallet-password

The SSL information for the Oracle Internet Directory connection.

enabled="boolean"

Required: true
Default: none
Valid Values: true or false

To enable SSL on the Oracle Internet Directory connection, set this attribute to true.

wallet-file="path"

Required: false
Default: none
Valid Values: A path

The path to a wallet file for authentication of the Oracle Internet Directory connection.

wallet-password="password"

Required: false
Default: none
Valid Values: A string

The password for the specified wallet-file.

timeout="depend-timeout"

Required: false
Default: 1200
Valid Values: An integer

The timeout attribute specifies in seconds how long OPMN will wait for a dependency check to complete. If the check takes longer than the configured timeout, then OPMN considers the check to have failed.

cache-timeout="cache-timeout"

Required: false
Default: 600
Valid Values: An integer

The cache-timeout attribute specifies how long in seconds OPMN will use the current "up" status for the dependency entry in the cache. If the last successful dependency check was within the prescribed number of seconds from the current check, then the dependency check is flagged as successful. Otherwise, OPMN performs another dependency check. The cache-timeout is only for the last successful check of the dependency. If the previous check failed, OPMN performs another access of the dependency check. A value of 0 indicates OPMN will always perform the check.

<OSSO>

Required: false

Default: none

Parents: [<dependencies>](#)

Attributes: host, port, URI, timeout, cache-timeout

Specifies the OracleAS Single Sign-On (osso) service to check.

host="hostname"

Required: true

Default: none

Valid Values: A string

The hostname for the OracleAS Single Sign-On connection.

port="port"

Required: true

Default: none

Valid Values: A port number

The port for the OracleAS Single Sign-On connection.

URI="uri"

Required: true

Default: none

Valid Values: A string

The URI for the OracleAS Single Sign-On connection.

<ssl>

Required: false

Default: none

Parents: [<OSSO>](#)

Attributes: enabled, wallet-file, wallet-password

The SSL information for the OracleAS Single Sign-On connection.

enabled="boolean"

Required: true

Default: none

Valid Values: true or false

To enable SSL on the OracleAS Single Sign-On connection, set this attribute to true.

wallet-file="path"

Required: true

Default: none

Valid Values: A path

The path to a wallet file for authentication of the OracleAS Single Sign-On connection.

wallet-password="password"

Required: false
 Default: none
 Valid Values: A string

The password for the specified wallet-file.

timeout="depend-timeout"

Required: false
 Default: 1200
 Valid Values: An integer

The timeout attribute specifies in seconds how long OPMN will wait for a dependency check to complete. If the check takes longer than the configured timeout, then OPMN considers the check to have failed.

cache-timeout="cache-timeout"

Required: false
 Default: 600
 Valid Values: An integer

The cache-timeout attribute specifies how long in seconds OPMN will use the current "up" status for the dependency entry in the cache. If the last successful dependency check was within the prescribed number of seconds from the current check, then the dependency check is flagged as successful. Otherwise, OPMN performs another dependency check. The cache-timeout is only for the last successful check of the dependency. If the previous check failed, OPMN performs another dependency check. A value of 0 indicates OPMN will always perform the check.

<managed-process>

Required: false
 Default: none
 Parents: **<dependencies>**

Attributes: ias-instance, ias-component, process-type, process-set, autostart, autostop, timeout, cache-timeout

Specifies the managed process to check. A process for process-type or process-set does not start unless the specified dependency managed process is alive. Circular dependencies are detected and rejected for local managed processes, but not for remote managed processes; this may result in a dependency check deadlock, which times out.

ias-instance="ias-instance-id"

Required: false
 Default: The ias-instance of the current process-type or process-set.
 Valid Values: A string

The ias-instance for the managed process dependency. If the specified ias-instance is not managed by the current OPMN, it is assumed to be a remote managed process dependency.

ias-component="ias-component-id"

Required: true
Default: none
Valid Values: A string

The ias-component for the managed process dependency.

process-type="process-type-id"

Required: true
Default: none
Valid Values: A string

The process-type-id for the managed process dependency.

process-set="process-set-id"

Required: true
Default: none
Valid Values: A string

The process-set-id for the managed process dependency.

autostart="boolean"

Required: false
Default: false
Valid Values: true or false

If the managed process dependency is not running when the check is performed, attempt to start it.

autostop="boolean"

Required: false
Default: false
Valid Values: true or false

When the managed process dependency is stopped, then stop the managed process. The attribute is always false for remote managed process dependencies.

timeout="depend-timeout"

Required: false
Default: 1200
Valid Values: An integer

The timeout attribute specifies, in seconds, how long OPMN will wait for a dependency check to complete. If the check takes longer than the configured timeout, then OPMN considers the check to have failed.

cache-timeout="cache-timeout"

Required: false
Default: 600
Valid Values: An integer

This parameter is only used for a process managed by a remote OPMN. The cache-timeout attribute specifies how long in seconds OPMN will use the current "up" status for the dependency entry in the cache. If the last timeout dependency check was within the prescribed number of seconds from the current check, then the dependency check is instantly flagged as successful, otherwise OPMN performs another dependency check. Note that the cache-timeout is only for the last successful check of the dependency, and if the previous check failed, OPMN performs another access of the dependency will be performed for this check. A value of 0 indicates OPMN will always perform the check.

Note: The cache-timeout is only for the last successful check of the dependency, and if the previous check failed, OPMN will perform another dependency check.

<process-type>

Required: true

Default: none

Parents: <ias-component>

Attributes: id, module-id, status, working-dir

A process-type is a grouping of process-sets that are supported by the same module.

id="process-type-id"

Required: true

Default: none

Valid Values: a string

The id attribute uniquely identifies this process-type within the ias-component.

module-id="module-id"

Required: true

Default: none

Valid Values: a string

The module-id attribute must map directly to the module-id element that supports this process-type.

status="state"

Required: false

Default: enabled

Valid Values: enabled or disabled

A process-type may be enabled, in which case OPMN parses all of its configured attributes and elements and allows requests to operate upon it, or disabled, in which case the process-type entry is completely ignored and treated as if it were not listed in opmn.xml.

working-dir="path"

Required: false

Default: None

Valid Values: A path

This path specifies the working directory set for managed processes created that belong to this process-type. If a process-set also defines a working-dir attribute, then that path takes precedence over the process-type path.

<event-scripts>

Required: false

Default: none

Parents: [<process-type>](#), [<process-set>](#)

Attributes: none

A configured event script is executed when a specific process related event has occurred. OPMN waits until the script completes or times out before proceeding with the next action for the process.

[Table 4–2](#) shows event script arguments.

Table 4–2 Event Script Arguments

Option Name	Option Argument	Description
-timeStamp	<time>	An integer value for the current time on the system (in seconds).
-instanceName	<instance-name>	The instance-name of the managed process.
-componentId	<component-id>	The component-id of the managed process.
-processType	<process-type-id>	The process-type of the managed process.
-processSet	<process-set-id>	The process-set of the managed process.
-processIndex	<index>	The process-index of the managed process.
-stderr ¹	<path>	The path for the stderr file pointer of the process.
-stdout ¹	<path>	The path for the stdout file pointer of the process. Note: this argument will only be given for a pre-start script if the start is part of a process restart request.
-reason	<reason>	A string indicating the reason script was executed. The http_request indicates the process action is the result of the user http request to OPMN. The non_http_request indicates the process action was initiated by OPMN itself.
-pid ²	<process-id>	The operating system integer value given for the process-id.
-startTime ²	<time>	An integer value for the system start time of the process (in seconds).

¹ This argument will only be given for a pre-start script if the start is part of a process restart request. The pre-start event is triggered only prior to performing a start. A restart operation may be composed of a stop operation followed by a start operation. A start operation can occur as an operation all by itself or as a sub-operation of a restart.

² This argument is only available with pre-stop or post-crash event scripts.

<pre-start>

Required: false
Default: none
Parents: [<event-scripts>](#)
Attributes: path

OPMN runs the pre-start script after any configured dependency checks have been performed (and passed) and before the process is actually started. The timeout for this script is the timeout value configured for starting the process itself, and any time consumed by the execution of this script counts towards the process start timeout. If the script times out, the process will not be started and any associated http request will fail.

Be cautious when you execute any OPMN process requests such as start, stop or restart within an event script. These requests are serialized at the process-set level. If the script invokes a request on a process-set on which the current request (or another already queued request) is operating, then the script will hang until it times out.

path="path"

Required: true
Default: none
Valid Values: A path to the executable script.

The path must specify either an executable program for which OPMN has execute permission, or a script file for which OPMN has both read and executable permission.

<pre-stop>

Required: false
Default: none
Parents: [<event-scripts>](#)
Attributes: path

OPMN runs the specified script before stopping the associated process. The timeout for this script is the value configured for stopping the process itself. Any time consumed by the execution of this script counts towards the process stop timeout. If the script times out, any associated http request will fail. However, OPMN will proceed with stopping the process.

Be cautious when you execute any OPMN process requests such as start, stop or restart. These requests are serialized at the process-set level. If the script invokes a request on a process-set on which the current request (or another already queued request) is operating, then the script will hang until it times out.

path="path"

Required: true
Default: none
Valid Values: A path to the executable script.

The path must specify either an executable program for which OPMN has execute permission, or a script file for which OPMN has both read and executable permission.

<post-crash>

Required: false

Default: none

Parents: [<event-scripts>](#)

Attributes: path

OPMN runs the specified script after the associated process has terminated unexpectedly. The timeout for this script is the timeout value configured for stopping the process itself. After the script has terminated OPMN schedules a replacement of the dead process.

Be cautious when you execute any OPMN process requests such as start, stop or restart. These requests are serialized at the process-set level. If the script invokes a request on a process-set on which the current request (or another already queued request) is operating, then the script will hang until it times out.

`path="path"`

Required: true

Default: none

Valid Values: A path to the executable script.

The path must specify either an executable program for which OPMN has execute permission, or a script file for which OPMN has both read and executable permission.

<start>

Required: false

Default: Refer to the values in the following paragraphs.

Parents: [<process-type>](#) , [<process-set>](#)

Attributes: timeout, retry

The start parameters for managed processes.

`timeout="timeout"`

Required: false

Default: 60

Valid Values: An integer

The timeout value in seconds for the start of a managed process.

`retry="num"`

Required: false

Default: 0

Valid Values: An integer

The number of consecutive attempts that will be made to start the process for a single request.

<stop>

Required: false

Default: Refer to the values in the following paragraphs.

Parents: [<process-type>](#) , [<process-set>](#)

Attributes: timeout

The stop parameters for managed processes.

timeout="timeout"

Required: false

Default: 30

Valid Values: An integer

The timeout value in seconds for the stopping a managed process.

<restart>

Required: false

Default: Refer to the values in the following paragraphs.

Parents: [<process-type>](#) , [<process-set>](#)

Attributes: timeout, retry

The restart parameters for managed processes.

timeout="timeout"

Required: false

Default: 90

Valid Values: An integer

The timeout value in seconds for the restart of a managed process.

retry="num"

Required: false

Default: 0

Valid Values: An integer

The number of consecutive attempts that will be made to restart the process for a single request.

<ping>

Required: false

Default: Refer to the values in the following paragraphs.

Parents: [<process-type>](#) , [<process-set>](#)

Attributes: timeout, retry, interval

The ping parameters for managed processes.

timeout="timeout"

Required: false

Default: 20

Valid Values: An integer

The timeout value in seconds for the ping of a managed process. Each module specifies a ping timeout.

retry="num"

Required: false

Default: 0

Valid Values: An integer

The number of consecutive ping failures that will be tolerated before the module declares the process unreachable and will restart it. Each module specifies ping retries.

interval="interval"

Required: false

Default: 20

Valid Values: An integer

The interval, in seconds, between each ping of a managed process.

<port>

Required: false

Default: none

Parents: [<process-type>](#), [<process-set>](#)

Attributes: id, range

OPMN provides a port management mechanism for modules to use. Each module uses the ports configured with id.

id="id"

Required: true

Default: none

Valid Values: A string

The id attribute identifies the range of ports for the process-type. Each module has its own list of required or optional port ids.

range="range"

Required: true

Default: none

Valid Values: A port range

The port range specifies which ports to use for the id.

Upon request from a module for a port number from the id, OPMN checks if a port in the range has been bound on the local system, and if it has not, it returns that port number back to the module. Syntax of the port range is a comma separated list of individual port numbers or a low-high range specification.

Examples:

Specify ports 5555, 6666, 7777, 8888, and 9999:

```
range="5555,6666,7777,8888,9999"
```

Specify ports 4000 through 4250 (inclusive):

```
range="4000-4250"
```

Specify ports 7000 through 7049, 7775, 7785, and 8050 through 8099:

```
range="7000-7049,7775,7785,8050-8099"
```

<process-set>

Required: true

Default: none

Parents: [<process-type>](#)

Attributes: id, restart-on-death, numprocs, status, working-dir

A process-set is the abstraction of a process within OPMN. All module-data, environment variables, and other configuration parameters are resolved into their final values at the process-set level.

`id="process-set-id"`

Required: true

Default: none

Valid Values: A string

The id attribute uniquely identifies this process-set within the process-type.

`restart-on-death="boolean"`

Required: true

Default: false

Valid Values: true or false

If a managed process terminates unexpectedly, that is, not stopped by a request, then OPMN will not automatically restart it. To enable automatic restarting of terminated managed processes set the attribute to true.

`numprocs="num"`

Required: true

Default: none

Valid Values: An integer

Specifies the number of processes for OPMN to start for this process-set.

`status="state"`

Required: false.

Default: enabled

Valid Values: enabled or disabled

A process-set may be enabled, in which case OPMN parses all of its configured attributes and elements and allows requests to operate upon it, or disabled, in which case the process-set entry is completely ignored and treated as if it were not even listed in opmn.xml.

`working-dir="path"`

Required: false.

Default: None

Valid Values: A path

This path specifies the working directory set for the managed processes created that belong to this process-set. \$ORACLE_HOME may be used.

OPMN will convert slashes in the working-dir value string to be those of the directory path separator character for the system on which OPMN is running (on UNIX each \ character is converted to / and on Windows each / is converted to \).

OPMN uses the ^ character as an escape character to disable slash conversion for the following character, and so ^/ on a Windows system will yield a / in the string. Specify two ^ characters if you need to specify the ^ character in the resultant string: ^^ yields ^.

5

Configuring Oracle HTTP Server

This chapter describes Oracle HTTP Server configuration in the OPMN `opmn.xml` file.

It features the following topics:

- [Oracle HTTP Server Process Module Configuration](#)
- [Oracle HTTP Server Minimum Configuration](#)
- [Oracle HTTP Server Complete Configuration](#)
- [Oracle HTTP Server Attribute Descriptions](#)

5.1 Oracle HTTP Server Process Module Configuration

The following lines load and identify the Oracle HTTP Server process module. Management of Oracle HTTP Server processes by the process module are identified by the module id.

```
<module path="ORACLE_HOME/opmn/lib/libopmnohs.so">
    <module-id="OHS" />
</module>
```

5.2 Oracle HTTP Server Minimum Configuration

The following lines represent the minimum configuration for Oracle HTTP Server. Default values are assigned to all other configuration elements and attributes for Oracle HTTP Server.

```
<ias-component id="HTTP_Server">
    <process-type id="HTTP_Server" module-id="OHS">
        <process-set id="HTTP_Server" numprocs="1"/>
    </process-type>
</ias-component>
```

5.3 Oracle HTTP Server Complete Configuration

The following lines show a complete configuration for Oracle HTTP Server. It contains all possible configuration elements and attributes for Oracle HTTP Server.

```
<ias-component id="HTTP_Server" status="enabled" id-matching="false">
    <process-type id="HTTP_Server" module-id="OHS">
        <process-set id="HTTP_Server" restart-on-death="true" numprocs=1>
            <module-data>
                <category id="start-parameters">
                    <data id="config-file" value="/myconfs/httpd.conf"/>
                    <data id="start-mode" value="ssl-disabled"/>
                    <data id="command-line" value="-D MyDefine"/>
                </category>
                <category id="ping-parameters">
                    <data id="ping-url" value="/" />
                </category>
                <category id="restart-parameters">
                    <data id="reverseping-timeout" value="345"/>
                    <data id="no-reverseping-failed-ping-limit" value="3"/>
                    <data id="reverseping-failed-ping-limit" value="6"/>
                </category>
            </module-data>
            <start timeout="300" retry="3"/>
            <stop timeout="300"/>
            <restart timeout="300"/>
            <ping timeout="30" interval="30"/>
        </process-set>
    </process-type>
</ias-component>
```

5.4 Oracle HTTP Server Attribute Descriptions

This section describes the attributes that are specific for Oracle HTTP Server.

The Oracle HTTP Server attributes are described with the following format:

- **Title:** This is the attribute name and value being defined. For example, `id="HTTP_Server"`.
- **Required:** This field defines whether or not the attribute is required in the component definition.
- **Default:** This defines the default value assigned to the attribute. The default value appears in the installed version of the `opmn.xml` file or is assigned internally if the attribute is not present.
- **Valid values:** This field defines the valid values for the attribute. For example, `HTTP_Server`.
- **Path:** This field defines in which elements the attribute can appear. For example, `ias-component/process-type/process-set`

`id="HTTP_Server"`

Required: true
 Default: none
 Valid values: `HTTP_Server`
 Path: `ias-component`
 Path: `ias-component/process-type`
 Path: `ias-component/process-type/process-set`

The `id` attribute is required and cannot be changed. The `id` must match the `targets.xml` entry or Application Server Control Console will not work.

`module-id="OHS"`

Required: true
 Default: none
 Path: `ias-component/process-type`

The `module-id` attribute defines the type of process. It associates the configuration with a process module.

`numprocs=1`

Required: true
 Default: none
 Valid values: 1
 Path: `ias-component/process-type/process-set`

The Number of OHS Instances to start. Only valid value is 1.

The `numprocs` attribute gives the number of Oracle HTTP Server instances to start. The only valid value is 1.

id="start-parameters"

Required: false
Default: none
Path: ias-component/process-type/process-set/module-data/category

The `start-parameters` category contains the parameters that are relevant for the startup of Oracle HTTP Server.

id="config-file"

Required: false
Default: \$ORACLE_HOME/Apache/Apache/conf/httpd.conf
Valid values: any full path to an existing configuration file
Path:
ias-component/process-type/process-set/module-data/category/
data

The `config-file` id is an start command option which specifies the `httpd.conf` for starting Oracle HTTP Server.

id="start-mode"

Required: false
Default: ssl-enabled
Valid values: ssl-enabled/ssl-disabled
Path: ias-component/process-type/process-set/module-data/
category/data

This option specifies whether Oracle HTTP Server will be started with `ssl` enabled

id="command-line"

Required: false
Default: none
Valid values: any valid command line options to Oracle HTTP Server
Path: ias-component/process-type/process-set/module-data/
category/data

This `id` option specifies extra command lines to append to the Oracle HTTP Server command line.

id="ping-parameters"

Required: false
Default: none
Path: ias-component/process-type/process-set/module-data/
category
The `ping` parameters category contains the parameters that configure how OPMN pings Oracle HTTP Server.

id="ping-url"

Required: false

Default: /

Valid values: the path portion of an url; for example: http://127.0.0.1/<path>

Path: ias-component/process-type/process-set/module-data/
category/data

The ping-url id specifies the URL at which OPMN pings Oracle HTTP Server.

id="restart-parameters"

Required: false

Default: none

Path: ias-component/process-type/process-set/module-data/
category

The restart-parameters category is used for defining parameters that will be used in death-detection.

id="reverseping-timeout"

Required: false

Default: 300 seconds

Valid values: Any reasonable timeout value

Path: ias-component/process-type/process-set/module-data/
category/data

The reverseping-timeout value is the maximum allowable time between two notifications arriving from an Oracle HTTP Server process. As part of death-detection, the Oracle HTTP Server module performs forward pings on the Oracle HTTP Server process. In the event that forward pings start failing, the reverse pings are taken into account in death-detection and Oracle HTTP Server processes are restarted.

id="no-reverseping-failed-ping-limit"

Required: false

Default: 1

Valid values: Any reasonable value that reflects the tolerance that OPMN should have for failed forward pings when reverse pings are also failing. This tolerance is used by OPMN to determine when the process should be declared as unresponsive and replaced.

Path:

ias-component/process-type/process-set/module-data/category/
data

This id module data element defines the tolerance for failed forward pings in the event that reverse pings are also not being received (within the timeout period specified by the reverseping-timeout data element). After the number of ping failures equals this limit, the process is deemed unresponsive and restarted by OPMN.

id="reverseping-failed-ping-limit"

Required: false

Default: 3

Valid values: Any reasonable value that reflects the tolerance that OPMN should have for failed forward pings when reverse pings are being received. This tolerance is used by OPMN to determine when the process should be declared as unresponsive and replaced.

Path:

`ias-component/process-type/process-set/module-data/category/
data`

This module_data element defines the tolerance for failed forward pings when reverse pings are succeeding. After the number of ping failures equals this limit, the process is deemed unresponsive and restarted by OPMN.

6

Configuring OC4J

This chapter describes OC4J configuration in the OPMN opmn.xml file.

It features the following topics:

- [OC4J Process Module Configuration](#)
- [OC4J Minimum Configuration](#)
- [OC4J Complete Configuration](#)
- [OC4J Attribute Descriptions](#)

6.1 OC4J Process Module Configuration

The following lines load and identify the OC4J process module. Management of OC4J processes by the process module are identified by the module id.

```
<module path="ORACLE_HOME/opmn/lib/libopmnoc4j.so">
  <module-id id="OC4J" />
</module>
```

6.2 OC4J Minimum Configuration

The following lines represent the minimum configuration for OC4J. Default values are assigned to all other configuration elements and attributes for OC4J.

```
<ias-component id="OC4J">
  <process-type id="home" module-id="OC4J">
    <port id="ajp" range="3301-3400" />
    <port id="rmi" range="3101-3200" />
    <port id="jms" range="3201-3300" />
    <process-set id="default-island" numprocs="1"/>
  </process-type>
</ias-component>
```

6.3 OC4J Complete Configuration

The complete configuration example in this section showcases the attributes of OC4J configuration that you can control. It contains all possible configuration elements and attributes that can be used with this component.

```
<ias-component id="OC4J" status="enabled" id-matching="false">
  <environment>
    <variable id="LD_LIBRARY_PATH" value="$ORACLE_HOME/lib"
      append="true" />
  </environment>
  <process-type id="home" module-id="OC4J">
    <port id="ajp" range="3301-3400" />
    <port id="rmi" range="3101-3200" />
    <port id="jms" range="3201-3300" />
  <process-set id="default-island" restart-on-death="true" numprocs="1">
    <module-data>
      <category id="start-parameters">
        <data id="java-options" value="-DTestVar=TestVal" />
        <data id="oc4j-options" value=" " />
        <data id="config-file" value="/my/config/dir/server.xml" />
        <data id="java-bin" value="/my/javalocation/jdk/bin/java" />
      </category>
      <category id="stop-parameters">
        <data id="java-options" value="-DTestVar=TestVal" />
      </category>
      <category id="restart-parameters">
        <data id="reverseping-timeout" value="345" />
        <data id="no-reverseping-failed-ping-limit" value="3" />
        <data id="reverseping-failed-ping-limit" value="6" />
      </category>
      <category id="urlping-parameters">
        <data id="/j2ee/servlet/Spy" value="200" />
      </category>
      <category id="security-parameters">
        <data id="wallet-file" value="file:/private/sanejane/ssl_cert/client_cert" />
        <data id="wallet-password" value="welcome1" />
      </category>
    </module-data>
  </process-set>
</ias-component>
```

```

    </category>
</module-data>
<start timeout="300" retry="3"/>
<stop timeout="300"/>
<restart timeout="300"/>
<ping timeout="30" interval="30"/>
</process-set>
</process-type>
</ias-component>

```

6.4 OC4J Attribute Descriptions

This section describes the attributes that are specific for OC4J. This section also provides attribute descriptions of the attributes.

The OC4J attributes are described with the following format:

- **Title:** This is the attribute name and value being defined. For example, **id="OC4J"**.
- **Required:** This field defines whether or not the attribute is required in the component definition.
- **Default:** This defines the default value assigned to the attribute. The default value appears in the installed version of the `opmn.xml` file or is assigned internally if the attribute is not present.
- **Valid values:** This field defines the valid values for the attribute. For example, OC4J.
- **Path:** This field defines in which elements the attribute can appear. For example, `ias-component`.

id="OC4J"

Required: true
 Default: none
 Valid values: OC4J
 Path: `ias-component`

The `id` name is required and cannot be changed. The `id` must match the entry in the `targets.xml` file or Application Server Control Console will not work.

module-id="OC4J"

Required: true
 Default: none
 Path: `ias-component/process-type`

The `module-id` associates the process with a module. For OC4J processes, this `id` has to match the `module-id` specified in the process module configuration for the OC4J module.

id="ajp"

Required: true
 Default: none
 Path: `ias-component/process-type/port`

A port element defining `ajp` port values is required.

id="rmi"

Required: true
Default: none
Path: ias-component/process-type/port

A port element defining rmi port values is required.

id="jms"

Required: true
Default: none
Path: ias-component/process-type/port

A port element defining jms port values is required.

range

Required: true
Default: none
Valid values: range of ports, individual port numbers or 0
Path: ias-component/process-type/port

This attribute is used to specify valid port ranges, comma separated list of ports or a mix of both. For port selection by the operating system to select ports, specify 0 and the OC4J process will use a port provided by the system.

<process-set>

Required: true
Default: none
Path: ias-component/process-type

For OC4J processes, the process-set element is administratively equivalent to an OC4J island.

id="start-parameters"

Required: false
Default: none
Path: ias-component/process-type/process-set/module-data/category

The start-parameters id is a category that collects all of the parameters that are relevant for the startup of an OC4J process.

id="java-options"

Required: false
Default: none
Valid values: any options acceptable to Java
Path: ias-component/process-type/process-set/module-data/category/data

OC4J requires that some java-options be passed to start and stop commands. These options are derived internally by OPMN, are not part of the opmn.xml configuration, and cannot be overridden. Additional java-options may be specified using this module data element.

id="oc4j-options"

Required: false
 Default: none
 Valid values: any options acceptable to the OC4J executable
 Path: ias-component/process-type/process-set/module-data/
 category/data

OC4J processes require options to be passed in as part of the start or stop commands to function correctly. These options cannot be overridden. In addition to these options, other options can be passed in via this module_data element. There is no default value for this data element.

id="config-file"

Required: false
 Default: \$ORACLE_HOME/j2ee/<process-type id>/config/server.xml
 Valid values: any full path to an existing configuration file
 Path: ias-component/process-type/process-set/module-data/
 category/data

The configuration file is an OC4J option in the start command. The default value for this data element is built from the \$ORACLE_HOME variable and OC4J instance name (process-type id).

id="java-bin"

Required: false
 Default: \$ORACLE_HOME/jdk/bin/java
 Valid values: Full path to java.exe
 Path: ias-component/process-type/process-set/module-data/
 category/data

The default value is the complete path to Java that is available in the installation. You can specify alternate paths to the Java executable. However, a valid version of Java will have to be used for the process to start up and work correctly.

id="stop-parameters"

Required: false
 Default: none
 Path: ias-component/process-type/process-set/module-data/
 category

The stop-parameters id is a category that includes all the parameters that are relevant for stopping an OC4J process.

id="restart-parameters"

Required: false
 Default: none
 Path: ias-component/process-type/process-set/module-data/
 category

The restart-parameters category is used for defining parameters that will be used in death-detection.

id="reverseping-timeout"

Required: false

Default: 300 seconds

Valid values: Any reasonable timeout value

Path: ias-component/process-type/process-set/module-data/category/data

The reverseping-timeout value is the maximum allowable time between two notifications arriving from an OC4J process. As part of death-detection, the OC4J module performs forward pings on the process also. In the event that forward pings start failing, the reverse pings are taken into account in death-detection and restart.

id="no-reverseping-failed-ping-limit"

Required: false

Default: 1

Valid values: Any value that reflects the tolerance that OPMN should have for failed forward pings when reverse pings are also failing. This tolerance is used by OPMN to determine when the process should be declared as unresponsive and replaced.

Path: ias-component/process-type/process-set/module-data/category/data

This module data element defines the tolerance for failed forward pings in the event that reverse pings are also not being received (within the timeout period specified by reverseping-timeout data element). After the number of ping failures equals this limit, the process is deemed unresponsive and restarted by OPMN.

id="reverseping-failed-ping-limit"

Required: false

Default: 3

Valid values: Any reasonable value that reflects the tolerance that OPMN should have for failed forward pings when reverse pings are being received. This tolerance is used by OPMN to determine when the process should be declared as unresponsive and replaced.

Path: ias-component/process-type/process-set/module-data/category/data

This module data element defines the tolerance for failed forward pings when reverse pings are succeeding. After the number of ping failures equals this limit, the process is deemed unresponsive and restarted by OPMN.

id="urlping-parameters"

Required: false

Default: Not Applicable

Valid values: Not Applicable

Path: ias-component/process-type/process-set/module-data/category

The "urlping-parameters" id enables users to specify URLs for ping operations as part of OC4J process ping operations. The data under this category consists of the URL and a valid HTTP return code. AJP13 protocol is used to directly connect to the OC4J process and the HTTP return code is validated against the configured code. If there are multiple URLs configured, failure in pinging any one of them will be considered a ping failure and the process will be restarted after the ping failures limit is exceeded.

id="/j2ee/servlet/Spy"

Required: false
 Default: Not Applicable
 Valid values: Any valid URL on the OC4J process.
 Path: ias-component/process-type/process-set/module-data/category/data

This is the URL in the OC4J process that will be pinged.

value="200"

Required: false
 Default: Not Applicable
 Valid values: Any valid HTTP return code.
 Path: ias-component/process-type/process-set/module-data/category/data

The following is the HTTP code that results from ping operations to the configured URL.

```
<category id="security-parameters">
  <data id="wallet-file" value="file:/private/sanejane/ssl_cert/client_cert"/>
  <data id="wallet-password" value="welcome1"/>
</category>
```

id="security-parameters"

Required: false
 Default: Not Applicable
 Valid values: Not Applicable
 Path: ias-component/process-type/process-set/module-data/category/

The OC4J process module can perform pings over SSL. The "security-parameters" id is a category that allows users to specify the wallet file and password for such communication.

id="wallet-file"

Required: false
 Default: Not Applicable
 Valid values: Not Applicable
 Path: ias-component/process-type/process-set/module-data/category/data

The data id whose value is the path to the wallet file (not including the filename).

value="file:/private/sanejane/ssl_cert/client_cert"

Required: false
 Default: Not applicable
 Valid values: Path to a wallet file (not including the filename).
 Path: ias-component/process-type/process-set/module-data/category/data

The path to the wallet file (not including the filename). The data in the wallet file is used in SSL authentication during pinging.

id="wallet-password"

Required: false

Default: Not applicable

Valid values: Not applicable

Path: ias-component/process-type/process-set/module-data/category/data

The data id that specifies the wallet password.

value ="welcome1"

Required: false

Default: Not applicable

Valid values: The valid wallet password.

Path: ias-component/process-type/process-set/module-data/category/data

This value specifies the password for the wallet.

Configuring Oracle Application Server Web Cache

This chapter describes Oracle Application Server Web Cache (OracleAS Web Cache) configuration in the OPMN opmn.xml file.

It features the following topics:

- [OracleAS Web Cache Process Module Configuration](#)
- [OracleAS Web Cache Minimum Configuration](#)
- [OracleAS Web Cache Complete Configuration](#)
- [OracleAS Web Cache Attribute Descriptions](#)

7.1 OracleAS Web Cache Process Module Configuration

The following lines load and identify the OracleAS Web Cache process module. Management of OracleAS Web Cache processes by the process module are identified by the module id.

```
<module path="ORACLE_HOME/opmn/lib/libopmnwc.so">
  <module-id id="WebCache" />
  <module-id id="WebCacheAdmin" />
</module>
```

7.2 OracleAS Web Cache Minimum Configuration

The following lines represent the minimum configuration for OracleAS Web Cache. Default values are assigned to all other configuration elements and attributes for OracleAS Web Cache.

```
<ias-component id="WebCache">
  <environment>
    <variable id="LD_LIBRARY_PATH" value="$ORACLE_HOME/lib:$ORACLE_HOME/opmn/lib"
      append="true"/>
  </environment>
  <process-type id="WebCache" module-id="WebCache">
    <process-set id="WebCache" numprocs="1"/>
  </process-type>
  <process-type id="WebCacheAdmin" module-id="WebCache-admin">
    <process-set id="WebCacheAdmin" restart-on-death="false" numprocs="1"/>
  </process-type>
</ias-component>
```

7.3 OracleAS Web Cache Complete Configuration

The following example represents the complete configuration for OracleAS Web Cache. It contains all possible configuration elements and attributes that can be used with OracleAS Web Cache.

```
<ias-component id="WebCache">
  id="WebCacheAdmin"
  <variable id="LD_LIBRARY_PATH" value="$ORACLE_HOME/lib:$ORACLE_HOME/opmn/lib"
    append="true"/>
  </environment>
  <process-type id="WebCache" module-id="WebCache">
    <process-set id="WebCache" restart-on-death="true" numprocs="1"/>
  </process-type>
  <process-type id="WebCacheAdmin" module-id="WebCache"-admin">
    <process-set id="WebCacheAdmin" restart-on-death="false" numprocs="1"/>
  </process-type>
</ias-component>
```

7.4 OracleAS Web Cache Attribute Descriptions

This section describes the attributes that are specific for OracleAS Web Cache.

The OracleAS Web Cache attributes are described with the following format:

- **Title:** This is the attribute name and value being defined. For example, `id="WebCache"`.
- **Required:** This field defines whether or not the attribute is required in the component definition.
- **Default:** This defines the default value assigned to the attribute. The default value appears in the installed version of the `opmn.xml` file or is assigned internally if the attribute is not present.
- **Valid values:** This field defines the valid values for the attribute. For example, `WebCache`.
- **Path:** This field defines in which elements the attribute can appear. For example, `ias-component`.

id="WebCache"

Required: true
 Default: none
 Valid values: WebCache
 Path: `ias-component`
 Path: `ias-component/process-type`
 Path: `ias-component/process-type/process-set`

This `id` is required and cannot be changed. The `id` must match the `targets.xml` entry or Application Server Control Console will not work.

id="WebCacheAdmin"

Required: true
 Default: none
 Valid values: WebCacheAdmin
 Path: `ias-component/process-type`
 Path: `ias-component/process-type/process-set`

The `id` name is required and cannot be changed. The `id` must match the `targets.xml` entry or elements and attributes will not work.

module-id="WebCache"

Required: true
 Default: none
 Valid values: WebCache, WebCache-admin
 Path: `ias-component/process-type`

The `module_id` defines whether the child `process-set` element defines a WebCache server process or a WebCache administrative process. At least one WebCache server process must be defined in the WebCache component.

restart-on-death

Required: true

Default: true

Valid values: true or false

Path: ias-component/process-type/process-set

The restart-on-death attribute defines whether the process will be restarted if it exits unexpectedly. By default after Oracle Application Server installation, this attribute is set to true for the OracleAS Web Cache server, and false for the OracleAS Web Cache administrative process.

numprocs="1"

Required: true

Default: none

Valid values: 1

Path: ias-component/process-type/process-set

This attribute gives the number of OracleAS Web Cache processes started for the process-set. The value must be set to 1.

8

Configuring Oracle Internet Directory

This chapter describes Oracle Internet Directory configuration in the OPMN `opmn.xml` file.

It features the following topics:

- [Oracle Internet Directory Process Module Configuration](#)
- [Oracle Internet Directory Minimum Configuration](#)
- [Oracle Internet Directory Complete Configuration](#)
- [Oracle Internet Directory Attribute Descriptions](#)

8.1 Oracle Internet Directory Process Module Configuration

The following lines load and identify the Oracle Internet Directory process module. Management of Oracle Internet Directory processes by the process module are identified by the module id.

```
<module path="$ORACLE_HOME/opmn/lib/libopmnoid.so">
    <module-id id="OID" />
</module>
```

8.2 Oracle Internet Directory Minimum Configuration

The following lines represent the minimum configuration for Oracle Internet Directory. Default values are assigned to all other configuration elements and attributes for Oracle Internet Directory.

```
<ias-component id="OID">
    <process-type id="OID" module-id="OID">
        <process-set id="OID" numprocs="1">
            <dependencies>
                <database db-connect-info="IASDB.US.ORACLE.COM"/>
            </dependencies>
            <module-data>
                <category id="oidctl-parameters">
                    <data id="connect" value="IASDB.US.ORACLE.COM"/>
                    <data id="startoidldapd" value="true"/>
                </category>
                <category id="oidmon-parameters">
                    <data id="connect" value="IASDB.US.ORACLE.COM"/>
                </category>
            </module-data>
        </process-set>
    </process-type>
</ias-component>
```

8.3 Oracle Internet Directory Complete Configuration

The following example represents the complete configuration for Oracle Internet Directory. It contains all possible configuration elements and attributes that can be used with Oracle Internet Directory.

```
<ias-component id="OID">
  <process-type id="OID" module-id="OID" working-dir="my_working_dir">
    <environment>
      <variable id="TNS_ADMIN" value="/ade/liyu_oid/oracle/work" />
    </environment>
    <process-set id="OID" numprocs="1">
      <dependencies>
        <database db-connect-info="IASDB.US.ORACLE.COM" />
      </dependencies>
      <module-data>
        <category id="oidmon-parameters">
          <data id="connect" value="IASDB.US.ORACLE.COM" />
          <data id="host" value="virtual_host_name" />
          <data id="sleep" value="20" />
          <data id="run-oidctl" value="true" />
        </category>
        <category id="oidctl-parameters">
          <data id="connect" value="IASDB.US.ORACLE.COM" />
          <data id="instance" value="1" />
          <data id="configset" value="1" />
          <data id="host" value="12.12.12.12" />
          <data id="flags" value="-p 389" />
          <data id="startoidldapd" value="true" />
          <data id="startoidrepId" value="true" />
          <data id="startodisrv" value="true" />
        </category>
      </module-data>
    </process-set>
  </process-type>
</ias-component>
```

8.4 Oracle Internet Directory Attribute Descriptions

This section describes the attributes that are specific for Oracle Internet Directory. This section also provides attribute descriptions of the attributes.

The Oracle Internet Directory attributes are described with the following format:

- **Title:** This is the attribute name and value being defined. For example, `id="OID"`.
- **Required:** This field defines whether or not the attribute is required in the component definition.
- **Default:** This defines the default value assigned to the attribute. The default value appears in the installed version of the `opmn.xml` file or is assigned internally if the attribute is not present.
- **Valid values:** This field defines the valid values for the attribute. For example, `OID`.
- **Path:** This field defines in which elements the attribute can appear. For example, `ias-component`.

id="OID"

Required: true
Default: none
Valid values: `OID`
Path: `ias-component`
Path: `ias-component/process-type`
Path: `ias-component/process-type/process-set`

This `id` name is required and cannot be changed. The name must match the entry `targets.xml` or Application Server Control Console will not work.

module-id="OID"

Required: true
Default: none
Path: `ias-component/process-type`

The `module-id` name defines the type of process and associates this configuration with a process module.

working-dir="my_working_dir"

Required: false
Default: `$ORACLE_HOME`
Path: `ias-component/process-type`

The `working-dir` attribute specifies where the `oidmon` process will run.

id="TNS_ADMIN"

Required: false
 Default: \$ORACLE_HOME/network/admin
 Path: ias-component/process-type/environment

Oracle Internet Directory requires the connection to the database by specifying a connect string. The string taken by Oracle Internet Directory is an alias. At runtime, the string is resolved using the `tnsnames.ora` file in the `$TNS_ADMIN` directory. By default, `$TNS_ADMIN` points to `$ORACLE_HOME/network/admin`. If `tnsnames.ora` exists under a different directory, the environment must be set to the different directory.

numprocs="1"

Required: true
 Default: none
 Valid values: 1
 Path: ias-component/process-type/process-set

This attribute gives the number of `oidmon` instances started for this process-set. There should be at most one instance running in one Oracle Application Server instance.

db-connect-info="IASDB.US.ORACLE.COM"

Required: true
 Default: none
 Valid values: The database connect string
 Path: ias-component/process-type/process-set

Oracle Internet Directory depends on the backend database. If the backend database is not running, `oidmon` and other Oracle Internet Directory server processes cannot start.

id="oidmon-parameters"

Required: false
 Default: none
 Path: ias-component/process-type/process-set/module-data/category

The `id` category specifies the startup parameters for `oidmon`.

id="connect"

Required: true
 Default: none
 Valid values: The database connect string alias
 Path: ias-component/process-type/process-set/module-data/category/data

The `id` data must be configured to start `oidmon`. The following command starts `oidmon`:

```
$ORACLE_HOME/bin/oidmon connect=<value> -opmnuid <uid> start
```

id="host"

Required: false

Default: none

Valid values: The virtual host name

Path: ias-component/process-type/process-set/module-data/category/data

The host data is configured by default at installation time. The following command starts oidmon:

```
$ORACLE_HOME/bin/oidmon connect=<value> host=<value> -opmnuid <uid> start
```

id="sleep"

Required: false

Default: 10 seconds

Valid values: The time interval in seconds at which oidmon monitors Oracle Internet Directory servers

Path: ias-component/process-type/process-set/module-data/category/data

If the sleep data is configured, the following command starts oidmon:

```
$ORACLE_HOME/bin/oidmon connect=<connect_value> sleep=<sleep_value> -opmnuid <uid> start
```

id="run-oidctl"

Required: false

Default: true

Valid values: true or false

Path: ias-component/process-type/process-set/module-data/category/data

OPMN manages oidmon. However, when starting and stopping oidmon, OPMN also attempts to start oidldap through oidctl by default.

If you do not want to start any process associated with oidctl, set this value to false.

id="instance"

Required: false

Default: 1

Valid values: The numerical value of the instance (>0 and <= 1000) to be started and stopped.

Path: ias-component/process-type/process-set/module-data/category/data

The instance value is required for using oidctl. If this id is not configured, the default value is 1. You can overwrite this default option by specifying a value.

id="configset"

Required: false

Default: 0

Valid values: The numerical value of the configuration set (>=0 and <= 1000)

Path: ias-component/process-type/process-set/module-data/category/data

The configset id is only needed to start oidldapd. If you want to start oidrepld or oidsrv, the assigned data is ignored. If configset is not configured, the default value is 0.

id="flags"

Required: false

Default: none

Path: ias-component/process-type/process-set/module-data/category/data

If flags is configured, it will overwrite the settings in the configset at runtime.

id="startoidldapd"

Required: false

Default: true

Valid values: true or false

Path: ias-component/process-type/process-set/module-data/category/data

If the id data is configured, the default start command for the server is:

```
$ORACLE_HOME/bin/oidctl connect=<connect_value> server=oidldapd instance=1
configset=0 start
```

id="startoidrepld"

Required: false

Default: false

Valid values: true or false

Path: ias-component/process-type/process-set/module-data/category/data

If the id data is configured, the default start command for the server is:

```
$ORACLE_HOME/bin/oidctl connect=<connect_value> server=oidrepld instance=1
configset=0 start
```

Note: This server must be configured before it can be brought up manually or by OPMN after installation.

id="startodisrv"

Required: false

Default: false

Valid values: true or false

Path: ias-component/process-type/process-set/module-data/
category/data

If the id data is configured, the default start command for the server is:

```
$ORACLE_HOME/bin/oidctl connect=<connect_value> server=odisrv instance=1  
configset=0 start
```

Configuring Oracle Application Server Port Tunnel

This chapter describes Oracle Application Server (OracleAS Port Tunnel) configuration in the OPMN `opmn.xml` file.

It features the following topics:

- [OracleAS Port Tunnel Process Module Configuration](#)
- [OracleAS Port Tunnel Minimum Configuration](#)
- [OracleAS Port Tunnel Complete Configuration](#)
- [OracleAS Port Tunnel Attribute Descriptions](#)

See Also: *Oracle HTTP Server Administrator's Guide*

9.1 OracleAS Port Tunnel Process Module Configuration

The following lines load and identify the OracleAS Port Tunnel process module. Management of OracleAS Port Tunnel processes by the process module are identified by the module id.

```
<module path="%ORACLE_HOME%/opmn/lib/libopmniaspt.so">
    <module-id id="IASPT" />
</module>
```

9.2 OracleAS Port Tunnel Minimum Configuration

The following lines represent the minimum configuration for OracleAS Port Tunnel. Default values are assigned to all other configuration elements and attributes for OracleAS Port Tunnel.

```
<ias-component id="IASPT">
    <process-type id="IASPT" module-id="IASPT">
        <process-set id="IASPT" numprocs="1"/>
    </process-type>
</ias-component>
```

9.3 OracleAS Port Tunnel Complete Configuration

The following example represents the complete configuration for OracleAS Port Tunnel. It contains all possible configuration elements and attributes that can be used with OracleAS Port Tunnel.

```
<module path="%ORACLE_HOME%/opmn/lib/libopmniaspt.so">
    <module-id id="IASPT" />
</module>
<ias-component id="IASPT" status="enabled" id-matching="false">
    <process-type id="IASPT" module-id="IASPT">
        <port id="ajp" range="6701-6703"/>
        <process-set id="IASPT" restart-on-death="true" id="ajp"/>
    </process-type>
</ias-component>
```

9.4 OracleAS Port Tunnel Attribute Descriptions

This section describes the attributes that are specific for OracleAS Port Tunnel.

The OracleAS Port Tunnel attributes are described with the following format:

- **Title:** This is the attribute name and value being defined. For example, `id="IASPT"`.
- **Required:** This field defines whether or not the attribute is required in the component definition.
- **Default:** This defines the default value assigned to the attribute. The default value appears in the installed version of the `opmn.xml` file or is assigned internally if the attribute is not present.
- **Valid values:** This field defines the valid values for the attribute. For example, `IASPT`.
- **Path:** This field defines in which elements the attribute can appear. For example, `ias-component`.

id="IASPT"

Required: true
 Default: none
 Valid values: IASPT
 Path: ias-component
 Path: ias-component/process-type
 Path: ias-component/process-set

The id name is required and cannot be changed. The id name must match the entry in the targets.xml file.

module-id="IASPT"

Required: true
 Default: none
 Path: ias-component/process-type

The module-id name defines the type of process and associates the configuration with a process module.

id="ajp"

Required: false
 Default: none
 Valid values: ajp
 Path: ias-component/process-type/port

The id value should be used together with range in port property to specify the ajp ports to be used by the OracleAS Port Tunnel server. If the id is specified, the port number configured in the iaspt.conf file is overwritten.

range="6701-6703"

Required: false
 Default: none
 Valid values: Any single port or a range of ports
 Path: ias-component/process-type/port

The range value should be used together with ajp in port property to specify the ajp ports to be used by OracleAS Port Tunnel servers.

numprocs="3"

Required: true
 Default: none
 Valid values: Any number
 Path: ias-component/process-type/process-set

This attribute tells how many OracleAS Port Tunnel server processes to be started. The ajp range should be configured in the port property if the value is 1. If the value is greater than 1, ajp range has to be configured to specify enough ports for each OracleAS Port Tunnel server process. Typically, the value is 1 port per process.

10

Configuring Oracle Application Server Wireless

This chapter describes Oracle Application Server Wireless (OracleAS Wireless) configuration in the OPMN `opmn.xml` file.

It features the following topics:

- [OracleAS Wireless Process Module Configuration](#)
- [OracleAS Wireless Minimum Configuration](#)
- [OracleAS Wireless Complete Configuration](#)
- [OracleAS Wireless Attribute Descriptions](#)

10.1 OracleAS Wireless Process Module Configuration

The following lines load and identify the OracleAS Wireless process module. Management of OracleAS Wireless processes by the process module are identified by the module id.

```
<module path="%ORACLE_HOME%/opmn/lib/libopmmoc4j.so">
  <module-id id="OC4J"/>
</module>
<module path="ORACLE_HOME/opmn/lib/libopmnwireless.so">
  <module-id id="alert"/>
  <module-id id="datafeeder"/>
  <module-id id="performance"/>
  <module-id id="messaging"/>
  <module-id id="notification"/>
</module>
```

10.2 OracleAS Wireless Minimum Configuration

The following lines represent the minimum configuration for OracleAS Wireless. Default values are assigned to all other configuration elements and attributes for OracleAS Wireless.

```
<ias-component id="wireless" status="enabled">
<environment>
  <variable id="WIRELESS_HOME" value="$ORACLE_HOME/wireless" append="false"/>
  <variable id="WIRELESS_LIB" value="$WIRELESS_HOME/lib" append="false"/>
  <variable id="JAVA13_HOME" value="$ORACLE_HOME/jdk" append="false"/>
  <variable id="JAVA_HOME" value="$JAVA13_HOME" append="false"/>
  <variable id="IAS_LIB" value="$ORACLE_HOME/lib" append="false"/>
  <variable id="CLASSPATH" value=". " append="true"/>
  <variable id="CLASSPATH" value="$JAVA13_HOME/jre/lib/rt.jar" append="true"/>
  <variable id="CLASSPATH" value="$WIRELESS_HOME/server/classes" append="true"/>
  <variable id="CLASSPATH" value="$WIRELESS_LIB/wireless.jar" append="true"/>
  <variable id="CLASSPATH" value="$ORACLE_HOME/jdbc/lib/classes12.jar" append="true"/>
  <variable id="CLASSPATH" value="$ORACLE_HOME/jdbc/lib/nls_charset12.jar" append="true"/>
  <variable id="CLASSPATH" value="$IAS_LIB/xmlparserv2.jar" append="true"/>
  <variable id="CLASSPATH" value="$ORACLE_HOME/jlib/jndi.jar" append="true"/>
  <variable id="CLASSPATH" value="$IAS_LIB/servlet.jar" append="true"/>
  <variable id="CLASSPATH" value="$IAS_LIB/activation.jar" append="true"/>
  <variable id="CLASSPATH" value="$IAS_LIB/mail.jar" append="true"/>
  <variable id="CLASSPATH" value="$WIRELESS_LIB/ssosdk902.jar" append="true"/>
  <variable id="CLASSPATH" value="$ORACLE_HOME/jlib/ldapjclnt9.jar" append="true"/>
  <variable id="CLASSPATH" value="$ORACLE_HOME/j2ee/home/ldap.jar" append="true"/>
  <variable id="CLASSPATH" value="$ORACLE_HOME/jlib/providerutil.jar" append="true"/>
  <variable id="CLASSPATH" value="$ORACLE_HOME/jlib/repository.jar" append="true"/>
  <variable id="CLASSPATH" value="$IAS_LIB/xschema.jar" append="true"/>
  <variable id="CLASSPATH" value="$ORACLE_HOME/sysman/webapps/emd/WEB-INF/lib/emd.jar" append="true"/>
  <variable id="CLASSPATH" value="$ORACLE_HOME/dcm/lib/dcm.jar" append="true"/>
  <variable id="CLASSPATH" value="$ORACLE_HOME/jlib/emPid.jar" append="true"/>
  <variable id="CLASSPATH" value="$WIRELESS_LIB/mwa.jar" append="true"/>
  <variable id="CLASSPATH" value="$ORACLE_HOME/jlib/netcfg.jar" append="true"/>
  <variable id="CLASSPATH" value="$ORACLE_HOME/j2ee/home/lib/http_client.jar" append="true"/>
  <variable id="CLASSPATH" value="$ORACLE_HOME/j2ee/home/lib/javax-ssl-1_2.jar" append="true"/>
  <variable id="CLASSPATH" value="$ORACLE_HOME/j2ee/home/lib/jssl-1_2.jar" append="true"/>
  <variable id="CLASSPATH" value="$ORACLE_HOME/jlib/ldapjclnt9.jar" append="true"/>
```

```

<variable id="CLASSPATH" value="$ORACLE_HOME/jlib/ojmisc.jar" append="true"/>
<variable id="CLASSPATH" value="$ORACLE_HOME/j2ee/home/jazn.jar" append="true"/>
<variable id="CLASSPATH" value="$ORACLE_HOME/j2ee/home/jaas.jar" append="true"/>
<variable id="CLASSPATH" value="$ORACLE_HOME/soap/lib/soap.jar" append="true"/>
</environment>
<process-type id="wireless_oc4j" module-id="OC4J">
    <port id="ajp" range="3301-3400" />
    <port id="rmi" range="3101-3200" />
    <port id="jms" range="3201-3300" />
    <process-set id="wireless-island" numprocs="1"/>
</process-type>
<process-type id="alert_server" module-id="alert">
    <stop timeout="300"/>
    <restart timeout="480"/>
    <process-set id="alert_instance_1" numprocs="1"/>
</process-type>
<process-type id="datafeeder_server" module-id="datafeeder">
    <stop timeout="300"/>
    <restart timeout="480"/>
    <process-set id="datafeeder_instance_1" numprocs="1"/>
</process-type>
<process-type id="performance_server" module-id="performance">
    <stop timeout="300"/>
    <restart timeout="480"/>
    <process-set id="performance_instance_1" numprocs="1"/>
</process-type>
<process-type id="messaging_server" module-id="messaging">
    <stop timeout="300"/>
    <restart timeout="480"/>
    <process-set id="messaging_instance_1" numprocs="1"/>
</process-type>
<process-type id="notification_server" module-id="notification">
    <stop timeout="300"/>
    <restart timeout="480"/>
    <process-set id="notification_instance_1" numprocs="1"/>
</process-type>
</ias-component>

```

10.3 OracleAS Wireless Complete Configuration

The following lines show a complete configuration for OracleAS Wireless. It contains all possible configuration elements and attributes for OracleAS Wireless.

```

<ias-component id="wireless" status="enabled">
    <environment>
        <variable id="WIRELESS_HOME" value="$ORACLE_HOME/wireless" append="false"/>
        <variable id="WIRELESS_LIB" value="$WIRELESS_HOME/lib" append="false"/>
        <variable id="JAVA13_HOME" value="$ORACLE_HOME/jdk" append="false"/>
        <variable id="JAVA_HOME" value="$JAVA13_HOME" append="false"/>
        <variable id="IAS_LIB" value="$ORACLE_HOME/lib" append="false"/>
        <variable id="CLASSPATH" value=". " append="true"/>
        <variable id="CLASSPATH" value="$JAVA13_HOME/jre/lib/rt.jar" append="true"/>
        <variable id="CLASSPATH" value="$WIRELESS_HOME/server/classes" append="true"/>
        <variable id="CLASSPATH" value="$WIRELESS_LIB/wireless.jar" append="true"/>
        <variable id="CLASSPATH" value="$ORACLE_HOME/jdbc/lib/classes12.jar" append="true"/>
        <variable id="CLASSPATH" value="$ORACLE_HOME/jdbc/lib/nls_charset12.jar" append="true"/>
        <variable id="CLASSPATH" value="$IAS_LIB/xmlparserv2.jar" append="true"/>
        <variable id="CLASSPATH" value="$ORACLE_HOME/jlib/jndi.jar" append="true"/>
        <variable id="CLASSPATH" value="$IAS_LIB/servlet.jar" append="true"/>
    </environment>

```

```

<variable id="CLASSPATH" value="$IAS_LIB/activation.jar" append="true"/>
<variable id="CLASSPATH" value="$IAS_LIB/mail.jar" append="true"/>
<variable id="CLASSPATH" value="$WIRELESS_LIB/ssosdk902.jar" append="true"/>
<variable id="CLASSPATH" value="$ORACLE_HOME/jlib/ldapjclnt9.jar" append="true"/>
<variable id="CLASSPATH" value="$ORACLE_HOME/j2ee/home/ldap.jar" append="true"/>
<variable id="CLASSPATH" value="$ORACLE_HOME/jlib/providerutil.jar" append="true"/>
<variable id="CLASSPATH" value="$ORACLE_HOME/jlib/repository.jar" append="true"/>
<variable id="CLASSPATH" value="$IAS_LIB/xschema.jar" append="true"/>
<variable id="CLASSPATH" value="$ORACLE_HOME/sysman/webapps/emd/WEB-INF/lib/emd.jar"
append="true"/>
<variable id="CLASSPATH" value="$ORACLE_HOME/dcm/lib/dcm.jar" append="true"/>
<variable id="CLASSPATH" value="$ORACLE_HOME/jlib/emPid.jar" append="true"/>
<variable id="CLASSPATH" value="$WIRELESS_LIB/mwa.jar" append="true"/>
<variable id="CLASSPATH" value="$ORACLE_HOME/jlib/netcfg.jar" append="true"/>
<variable id="CLASSPATH" value="$ORACLE_HOME/j2ee/home/lib/http_client.jar"
append="true"/>
<variable id="CLASSPATH" value="$ORACLE_HOME/j2ee/home/lib/javax-ssl-1_2.jar"
append="true"/>
<variable id="CLASSPATH" value="$ORACLE_HOME/j2ee/home/lib/jssl-1_2.jar" append="true"/>
<variable id="CLASSPATH" value="$ORACLE_HOME/jlib/ldapjclnt9.jar" append="true"/>
<variable id="CLASSPATH" value="$ORACLE_HOME/jlib/ojmisc.jar" append="true"/>
<variable id="CLASSPATH" value="$ORACLE_HOME/j2ee/home/jazn.jar" append="true"/>
<variable id="CLASSPATH" value="$ORACLE_HOME/j2ee/home/jaas.jar" append="true"/>
<variable id="CLASSPATH" value="$ORACLE_HOME/soap/lib/soap.jar" append="true"/>
</environment>
<process-type id="wireless_oc4j" module-id="OC4J">
<port id="ajp" range="3301-3400" />
<port id="rmi" range="3101-3200" />
<port id="jms" range="3201-3300" />
<process-set id="wireless-island" numprocs="1" />
</process-type>
<process-type id="alert_server" module-id="alert" working-dir="%ORACLE_HOME%/wireless">
<module-data>
    <category id="start-parameters">
        <data id="java-bin" value="/my/path/to/java/exec"/>
        <data id="java-parameters" value="-Doracle.wireless.instance=my_alert_instance_name
-Dyet_another_definition=yet_another_value"/>
        <data id="class-name" value="MyJavaStartClass"/>
        <data id="application-parameters" value="my_application_parameters"/>
    </category>
    <category id="stop-parameters">
        <data id="java-bin" value="/my/path/to/java/exec"/>
        <data id="java-parameters" value="-Doracle.wireless.instance=my_alert_instance_name
-Dyet_another_definition=yet_another_value"/>
        <data id="class-name" value="MyJavaStopClass"/>
        <data id="application-parameters" value="my_application_parameters"/>
    </category>
</module-data>
<stop timeout="300"/>
<restart timeout="480"/>
<process-set id="alert_instance_1" numprocs="1" />
</process-type>
<process-type id="datafeeder_server" module-id="datafeeder" working-dir="ORACLE_HOME/wireless">
<module-data>
    <category id="start-parameters">
        <data id="java-bin" value="/my/path/to/java/exec"/>
        <data id="java-parameters" value="-Doracle.wireless.instance=my_datafeeder
_instance_name
-Dyet_another_definition=yet_another_value"/>
        <data id="class-name" value="MyJavaStartClass"/>
    </category>
</module-data>

```

```

        <data id="application-parameters" value="my_application_parameters"/>
    </category>
<category id="stop-parameters">
    <data id="java-bin" value="/my/path/to/java/exec"/>
    <data id="java-parameters" value="-Doracle.wireless.instance=
my_datafeeder_instance_name -Dyet_another_definition=yet_another_value"/>
    <data id="class-name" value="MyJavaStopClass"/>
    <data id="application-parameters" value="my_application_parameters"/>
</category>
</module-data>
<stop timeout="300"/>
<restart timeout="480"/>
<process-set id="datafeeder_instance_1" numprocs="1"/>
</process-type>
<process-type id="performance_server" module-id="performance"
working-dir="ORACLE_HOME/wireless">
    <module-data>
        <category id="start-parameters">
            <data id="java-bin" value="/my/path/to/java/exec"/>
            <data id="java-parameters"
value="-Doracle.wireless.instance=my_performance_instance_name
-Dyet_another_definition=yet_another_value"/>
            <data id="class-name" value="MyJavaStartClass"/>
            <data id="application-parameters" value="my_application_parameters"/>
        </category>
        <category id="stop-parameters">
            <data id="java-bin" value="/my/path/to/java/exec"/>
            <data id="java-parameters"
value="-Doracle.wireless.instance=
my_performance_instance_name-Dyet_another_definition=yet_another_value"/>
            <data id="class-name" value="MyJavaStopClass"/>
            <data id="application-parameters" value="my_application_parameters"/>
        </category>
    </module-data>
    <stop timeout="300"/>
    <restart timeout="480"/>
    <process-set id="performance_instance_1" numprocs="1"/>
</process-type>
<process-type id="messaging_server"
module-id="messaging"
working-dir="ORACLE_HOME/wireless">
    <module-data>
        <category id="start-parameters">
            <data id="java-bin" value="/my/path/to/java/exec"/>
            <data id="java-parameters"
value="-Doracle.wireless.instance=my_messaging_instance_name
-Dyet_another_definition=yet_another_value"/>
            <data id="class-name" value="MyJavaStartClass"/>
            <data id="application-parameters" value="my_application_parameters"/>
        </category>
        <category id="stop-parameters">
            <data id="java-bin" value="/my/path/to/java/exec"/>
            <data id="java-parameters" value="-Doracle.wireless.instance=
my_messaging_instance_name -Dyet_another_definition=yet_another_value"/>
            <data id="class-name" value="MyJavaStopClass"/>
            <data id="application-parameters" value="my_application_parameters"/>
        </category>
    </module-data>
    <stop timeout="300"/>
    <restart timeout="480"/>

```

```
<process-set id="messaging_instance_1" numprocs="1"/>
</process-type>
<process-type id="notification_server" module-id="notification" working-dir=
"ORACLE_HOME/wireless">
<module-data>
  <category id="id=start-parameters">
    <data id="java-bin" value="/my/path/to/java/exec"/>
    <data id="java-parameters" value="-Doracle.wireless.instance=
my_notification_instance_name -Dyet_another_definition=yet_another_value"/>
    <data id="class-name" value="MyJavaStartClass"/>
    <data id="application-parameters" value="my_application_parameters"/>
  </category>
  <category id="stop-parameters">
    <data id="java-bin" value="/my/path/to/java/exec"/>
    <data id="java-parameters" value="-Doracle.wireless.instance=
my_notification_instance_name -Dyet_another_definition=yet_another_value"/>
    <data id="class-name" value="MyJavaStopClass"/>
    <data id="application-parameters" value="my_application_parameters"/>
  </category>
</module-data>
<stop timeout="300"/>
<restart timeout="480"/>
<process-set id="notification_instance_1" numprocs="1"/>
</process-type>
</ias-component>
```

10.4 OracleAS Wireless Attribute Descriptions

This section describes the attributes that are specific for OracleAS Wireless.

The OracleAS Wireless attributes are described with the following format:

- **Title:** This is the attribute name and value being defined. For example, `id="wireless"`.
- **Required:** This field defines whether or not the attribute is required in the component definition.
- **Default:** This defines the default value assigned to the attribute. The default value appears in the installed version of the `opmn.xml` file or is assigned internally if the attribute is not present.
- **Valid values:** This field defines the valid values for the attribute. For example, `wireless`.
- **Path:** This field defines in which elements the attribute can appear. For example, `ias-component`.

id="wireless"

Required: true
Default: none
Valid values: wireless
Path: ias-component

This `id` is required and cannot be changed. The name must match the `targets.xml` entry or elements and attributes will not work.

module-id="alert"

Required: true
 Default: none
 Path: ias-component/process-type

This module-id defines the type of process and associates this configuration with a process module. In [Section 10.1](#), the five OracleAS Wireless server types are associated with OracleAS Wireless process module.

module-id="datafeeder"

Required: true
 Default: none
 Path: ias-component/process-type

This module-id defines the type of process and associates this configuration with a process module.

module-id="performance"

Required: true
 Default: none
 Path: ias-component/process-type

This module-id defines the type of process and associates this configuration with a process module.

module-id="messaging"

Required: true
 Default: none
 Path: ias-component/process-type

This module-id defines the type of process and associates this configuration with a process module.

module-id="notification"

Required: true
 Default: none
 Path: ias-component/process-type

This module-id defines the type of process and associates this configuration with a process module.

numprocs="1"

Required: true
 Default: none
 Valid values: 1
 Path: ias-component/process-type/process-set

This attribute gives the number of OracleAS Wireless instances started for this process-set. The OracleAS Wireless module does not permit more than one instance with the same process-set. To start more than one processes for the same server type, put one process-set entry with one unique process-set id for each process.

For example:

```
<process-type id="datafeeder_server" module-id="datafeeder">
  <process-set id="datafeeder_instance_1" numprocs="1"/>
  <process-set id="datafeeder_instance_2" numprocs="1"/>
</process-type>
```

Do not create the following:

```
<process-type id="datafeeder_server" module-id="datafeeder">
  <process-set id="datafeeder_instance_1" numprocs="2"/>
</process-type>
```

id="start-parameters"

Required: false

Default: none

Path: ias-component/process-type/process-set/module-data/
category

This id indicates the configuration block for start-parameters category. This category contains data related to starting the module.

id="java-bin"

Required: false

Default: \$ORACLE_HOME/jdk/bin/java

Valid values: Full path to java.exe

Path: ias-component/process-type/process-set/module-data/
category/data

You can specify alternate paths to the Java executable. However, a valid version of Java will have to be used for the process to start up and work correctly.

id="java-parameters"

Required: false

Default: None

Valid values: Any valid Java parameters

Path: ias-component/process-type/process-set/module-data/
category/data

This id allows you to specify any valid Java parameters. All specified Java parameters will be appended to the default values. OPMN derives some java parameters internally and additional parameters can be added with this attribute.

id="class-name"

Required: false

Default: oracle.panama.sysmgmt.server.PanamaServer for start and
oracle.panama.sysmgmt.client.manager for stop

Valid values: Any valid Java class names

Path: ias-component/process-type/process-set/module-data/
category/data

This id allows you to specify a Java class name other than the default ones to start or stop a OracleAS Wireless process. Oracle recommends not changing this parameter; it is for internal use only.

id="application-parameters"

Required: false

Default: none

Valid values: any valid Java application parameters

Path: ias-component/process-type/process-set/module-data/
category/data

This id allows you to put more application parameters in addition to the default one. The default one tells which OracleAS Wireless process-type to be started.

id="stop-parameters"

Required: false

Default: none

Path: ias-component/process-type/process-set/module-data/
category

This id indicates the configuration block for stop-parameters category. In this category, some stop related module data can be specified.

See Also: [id="start-parameters"](#)

stop timeout="300"

Required: false

Default: 180

Valid values: A timeout value in seconds

Path: ias-component/process-type/

This parameter notifies OPMN how long it should wait for the process to terminate. The default value is 180 seconds. However, when stopping OracleAS Wireless processes, database access is required and it may take up to 5 minutes to signal a process to terminate.

restart timeout="480"

Required: false

Default: 180

Valid values: A timeout value in seconds

Path: ias-component/process-type/

This parameter notifies OPMN the waiting period for a OracleAS Wireless process to be restarted. The default value is 180 seconds. However, a restart action is actually a stop and start, so the total value should be 180 (for start)+300 (for stop)=480 seconds.

Configuring Oracle Application Server ProcessConnect

This chapter describes Oracle Application Server OracleAS ProcessConnect configuration in the OPMN opmn.xml file.

It features the following topics:

- [OracleAS ProcessConnect Module Configuration](#)
- [OracleAS ProcessConnect Minimum Configuration](#)
- [OracleAS ProcessConnect Complete Configuration](#)
- [OracleAS ProcessConnect Attribute Descriptions](#)

11.1 OracleAS ProcessConnect Module Configuration

The following lines load and identify the OracleAS ProcessConnect process module. Management of OracleAS ProcessConnect processes by the process module are identified by the module id.

```
<module path="%ORACLE_HOME%/opmn/lib/libopmnip.so">
  <module-id id="integration-manager"/>
  <module-id id="adapter-framework"/>
</module>
```

11.2 OracleAS ProcessConnect Minimum Configuration

The following lines represent the minimum configuration for OracleAS ProcessConnect. Default values are assigned to all other configuration elements and attributes for OracleAS ProcessConnect.

```
<ias-component id="ProcessConnect">
  <environment>
    <variable id="LANG" value="en_US" append="false"/>
    <variable id="THREADS_FLAG" value="native" append="false"/>
    <variable id="CLASSPATH" value="$ORACLE_HOME/rdbms/jlib/xdb_g.jar" append="true"/>
    <variable id="CLASSPATH" value="$ORACLE_HOME/jlib/rts2.jar" append="true"/>
    <variable id="CLASSPATH" value="$ORACLE_HOME/j2ee/home/oc4j.jar" append="true"/>
    <variable id="CLASSPATH" value="$ORACLE_HOME/j2ee/home/lib/http_client.jar"
      append="true"/>
    <variable id="CLASSPATH" value="$ORACLE_HOME/j2ee/home/lib/aqapi.jar" append="true"/>
    <variable id="CLASSPATH" value="$ORACLE_HOME/ip/admin" append="true"/>
    <variable id="CLASSPATH" value="$ORACLE_HOME/ip/work" append="true"/>
    <variable id="CLASSPATH" value="$ORACLE_HOME/ip/config" append="true"/>
    <variable id="CLASSPATH" value="$ORACLE_HOME/ip/lib/ip.jar" append="true"/>
  </environment>
  <process-type id="IntegrationManager" module-id="integration-manager">
    <port id="dms" range="8301-8401"/>
    <process-set id="im_instance1" numprocs="1"/>
  </process-type>
  <process-type id="AdapterFramework" module-id="adapter-framework">
    <port id="dms" range="8301-8401"/>
    <process-set id="af_instance1" numprocs="1"/>
  </process-type>
</ias-component>
```

11.3 OracleAS ProcessConnect Complete Configuration

The following lines show a complete configuration for OracleAS ProcessConnect. It contains all possible configuration elements and attributes for OracleAS ProcessConnect.

```

<ias-component id="ProcessConnect" status="enabled">
  <module-data>
    <category id="start-parameters">
      <data id="java-bin" value="/my/path/to/java/exec"/>
      <data id="java-parameters" value="-ms8m -mx512m"/>
      <data id="class-name" value="MyJavaStartClass"/>
    </category>
    <category id="stop-parameters">
      <data id="java-bin" value="/my/path/to/java/exec"/>
      <data id="java-parameters" value="-ms8m -mx512m"/>
      <data id="class-name" value="MyJavaStopClass"/>
    </category>
    <category id="ip-parameters">
      <data id="dms-port-property-name" value="my.dms.port.property"/>
      <data id="log-dir" value="/my/path/to/ip/log/dir"/>
      <data id="start-command" value="-mystartstring"/>
      <data id="shutdown-command" value="-myshutdownstring"/>
      <data id="im-string" value="MyIntegrationManagerString"/>
      <data id="af-string" value="My Adapter Framework String"/>
      <data id="im-class-name" value="MyIMClass"/>
      <data id="af-class-name" value="MyAFClass"/>
    </category>
  </module-data>
  <environment>
    <variable id="LANG" value="en_US" append="false"/>
    <variable id="THREADS_FLAG" value="native" append="false"/>
    <variable id="CLASSPATH" value="$ORACLE_HOME/rdbms/jlib/xdb_g.jar"
      append="true"/>
    <variable id="CLASSPATH" value="$ORACLE_HOME/jlib/rts2.jar" append="true"/>
    <variable id="CLASSPATH" value="$ORACLE_HOME/j2ee/home/oc4j.jar" append="true"/>
    <variable id="CLASSPATH" value="$ORACLE_HOME/j2ee/home/lib/http_client.jar"
      append="true"/>
    <variable id="CLASSPATH" value="$ORACLE_HOME/j2ee/home/lib/aqapi.jar"
      append="true"/>
    <variable id="CLASSPATH" value="$ORACLE_HOME/ip/admin" append="true"/>
    <variable id="CLASSPATH" value="$ORACLE_HOME/ip/work" append="true"/>
    <variable id="CLASSPATH" value="$ORACLE_HOME/ip/config" append="true"/>
    <variable id="CLASSPATH" value="$ORACLE_HOME/ip/lib/ip.jar" append="true"/>
  </environment>
  <process-type id="IntegrationManager" module-id="integration-manager">
    <port id="dms" range="8301-8401"/>
    <id="im_instance1" numprocs="1" />
  </process-type>
  <process-type id="AdapterFramework" module-id="adapter-framework">
    <port id="dms" range="8301-8401"/>
    <process-set id="af_instance1" numprocs="1" />
  </process-type>
</ias-component>
```

11.4 OracleAS ProcessConnect Attribute Descriptions

This section describes the attributes that are specific for OracleAS ProcessConnect. This section also provides attribute descriptions of the attributes.

The OracleAS ProcessConnect attributes are described with the following format:

- **Title:** This is the attribute name and value being defined. For example, `id="ProcessConnect"`.
- **Required:** This field defines whether or not the attribute is required in the component definition.
- **Default:** This defines the default value assigned to the attribute. The default value appears in the installed version of the `opmn.xml` file or is assigned internally if the attribute is not present.
- **Valid values:** This field defines the valid values for the attribute. For example, `ProcessConnect`.
- **Path:** This field defines in which elements the attribute can appear. For example, `ias-component`.

id="ProcessConnect"

Required: true
Default: none
Valid values: `ProcessConnect`
Path: `ias-component`

This `id` is required and cannot be changed. The name must match the `targets.xml` entry or elements and attributes will not work.

id="start-parameters"

Required: false
Default: none
Path: `ias-component/module-data/category`

This `id` indicates the configuration block for the `start-parameters` category. In this category, some start-related module data can be specified.

id="stop-parameters"

Required: false
Default: none
Path: `ias-component/module-data/category`

This `id` indicates the configuration block for `stop-parameters` category. In this category, some stop-related module data can be specified.

id="ip-parameters"

Required: false
Default: none
Path: `ias-component/module-data/category`

This `id` indicates the configuration block for the `ip-parameters` category. In this category, some ProcessConnect-related module data can be specified.

id="java-bin"

Required: false
 Default: \$ORACLE_HOME/jdk/bin/java
 Valid values: Full path to java.exe
 Path: ias-component/module-data/category

You can specify alternate paths to the Java executable. However, a valid version of Java will have to be used for the process to start up and work correctly.

id="java-parameters"

Required: false
 Default: -ms8m -mx512m -D<IM|AF>
 -Doracle.dms.httpd.port.start=<dms_port>
 -Doracle.opmn.uid=<opmn_uid> -Doracle.ons.oraclehome=<ORACLE_HOME>
 Valid values: Any valid Java parameters
 Path: ias-component/module-data/category/data

You can specify any valid Java parameters for the Java command line. If a parameter matches one in the default parameter list, it will override the default.

id="class-name"

Required: false
 Default: oracle.tip.runtime.admin.ProcessManager
 Valid values: A valid Java class name
 Path: ias-component/module-data/category/data

You can specify any valid Java class name that will be used for starting or stopping OracleAS ProcessConnect processes. This attribute is for internal use only and must not be changed by the user.

id="dms-port-property-name"

Required: false
 Default: oracle.dms.httpd.port.start
 Valid values: A valid Java property name
 Path: ias-component/module-data/category/data

You can specify a Java property name that will be used to pass an OPMN assigned DMS port value to OracleAS ProcessConnect processes. In turn, OracleAS ProcessConnect processes will read this property to get the DMS port. This name should match the property name used by OracleAS ProcessConnect processes for retrieving the DMS port.

id="log-dir"

Required: false
 Default: \$ORACLE_HOME/ip/log
 Valid values: The file path where OracleAS ProcessConnect processes store log files
 Path: ias-component/module-data/category/data

You can specify a directory where OracleAS ProcessConnect processes will store their log files.

id="start-command"

Required: false

Default: -start

Valid values: A valid OracleAS ProcessConnect command for starting OracleAS ProcessConnect processes

Path: ias-component/module-data/category/data

This is an OracleAS ProcessConnect process management specific command to be passed into OracleAS ProcessConnect process management class. It should not be changed.

id="shutdown-command"

Required: false

Default: -shutdown

Valid values: A valid OracleAS ProcessConnect specific command for stopping OracleAS ProcessConnect processes

Path: ias-component/module-data/category/data

This is an OracleAS ProcessConnect process management specific command to be passed into OracleAS ProcessConnect process management class. It should not be changed.

id="im-string"

Required: false

Default: IntegrationManager

Valid values: A valid OracleAS ProcessConnect specific process management parameter

Path: ias-component/module-data/category/data

This is an OracleAS ProcessConnect process management specific parameter to be passed into OracleAS ProcessConnect process management class for Integration Manager process. It should not be changed.

id="af-string"

Required: false

Default: "Adapter Framework"

Valid values: A valid OracleAS ProcessConnect specific process management parameter

Path: ias-component/module-data/category/data

This is an OracleAS ProcessConnect process management specific parameter to be passed into OracleAS ProcessConnect process management class for Adapter Framework process. It should not be changed.

id="im-class-name"

Required: false

Default: oracle.tip.runtime.im.IntegrationManager

Valid values: A valid Java class name

Path: ias-component/module-data/category/data

This parameter tells OracleAS ProcessConnect process management class which class name will be used to instantiate Integration Manager class. It should not be changed.

id="af-class-name"

Required: false
Default: oracle.tip.adapter.fw.bootstrap.AFInstantiator
Valid values: A valid Java class name
Path: ias-component/module-data/category/data

This parameter tells OracleAS ProcessConnect process management class which class name will be used to instantiate Adapter Framework class. It should not be changed.

module-id="integration-manager"

Required: true
Default: none
Valid values: integration-manager
Path: ias-component/process-type

This id defines the type of process and associates the configuration with a process module.

module-id="adapter-framework"

Required: true
Default: none
Valid values: adapter-framework
Path: ias-component/process-type

This id defines the type of process and associates the configuration with a process module.

id="dms"

Required: true
Default: none
Valid values: dms
Path: ias-component/process-type/port

This id specifies the range of ports for DMS inside each OracleAS ProcessConnect process.

range="8301-8401"

Required: true
Default:
Valid values: A valid port number or a range of ports
Path: ias-component/process-type/port

You should use this parameter to specify a port or a range of ports to be used by DMS inside each OracleAS ProcessConnect process.

id="im_instance1"

Required: true

Default: none

Valid values: An OracleAS ProcessConnect process instance name

Path: ias-component/process-type/process-set

This parameter lists the process instance name for the process-set. This instance name will be passed into the OracleAS ProcessConnect process management class for starting or stopping OracleAS ProcessConnect processes.

numprocs="1"

Required: true

Default: none

Valid values: 1

Path: ias-component/process-type/process-set

This attribute gives the number of OracleAS ProcessConnect instances started for the process-set. The OracleAS ProcessConnect module does not permit more than one instance with the same process-set. To start more than one processes for the same server type, put one process-set entry with one unique process-set instance id for each process.

12

Configuring Oracle Application Server Reports Services

This chapter describes Oracle Application Server Reports Services (OracleAS Reports Services) configuration in the OPMN `opmn.xml` file.

It features the following topics:

- [Introduction](#)
- [OracleAS Reports Services Process Module Configuration](#)
- [OracleAS Reports Services Minimum Configuration](#)
- [OracleAS Reports Services Complete Configuration](#)
- [OracleAS Reports Services Attribute Descriptions](#)

12.1 Introduction

The OracleAS Reports Services server can run in two modes:

- in-process
- standalone

The in-process OracleAS Reports Services server runs inside an OC4J instance and is managed by the OC4J module. In this mode, the OracleAS Reports Services module does not have to be configured. The OracleAS Reports Services server configuration is part of the OC4J instance configuration and is handled by the OC4J module.

The standalone OracleAS Reports Services server is managed by the OracleAS Reports Services module. The OracleAS Reports Services process module must be configured as shown in the [Section 12.2, "OracleAS Reports Services Process Module Configuration"](#). There is also an Oracle Application Server component configured to represent the standalone OracleAS Reports Services server.

12.2 OracleAS Reports Services Process Module Configuration

The following lines load and identify the OracleAS Reports Services process module. Management of OracleAS Reports Services processes by the process module are identified by the module id.

```
<module path="/private/oraclehome/opmn/lib/libopmnreports.so">
  <module-id id="ReportsModule" />
</module>
```

12.3 OracleAS Reports Services Minimum Configuration

This section describes the minimum configuration for OracleAS Reports Services. It features the following topics:

- [In-process OracleAS Reports Services Server](#)
- [Standalone OracleAS Reports Services Server](#)

12.3.1 In-process OracleAS Reports Services Server

The following example shows the configuration required for the in-process OracleAS Reports Services server. The lines should be placed within the configuration for the appropriate OC4J instance. The configuration directives are handled by the OC4J module.

```
<ias-component ....>
  <process-type ....>
    <module-data>
      <category id="urlping-parameters">
        <data id="/reports/rwervlet/pingserver?start=auto" value="200"/>
      </category>
    </module-data>
    <process-set .../>
  </process-type>
</ias-component>
```

12.3.2 Standalone OracleAS Reports Services Server

The following example represents the minimum configuration for a standalone OracleAS Reports Services server component. Reasonable default values are assigned to all other configuration elements and attributes that can be used with this component.

```
<ias-component id="ReportsServer1">
  <process-type id="ReportsServer1" module-id="ReportsModule">
    <process-set id="ReportsServer1" numprocs="1">
      <environment>
        <variable id="PATH" value="Put Path_value from env here" append="true"/>
      </environment>
    </process-set>
  </process-type>
</ias-component>
```

12.4 OracleAS Reports Services Complete Configuration

The following example represents the complete configuration for OracleAS Reports Services. It contains all possible configuration elements and attributes that can be used with this component. This section features the following topics:

- [In-process OracleAS Reports Services Server](#)
- [Standalone OracleAS Reports Services Server](#)

12.4.1 In-process OracleAS Reports Services Server

The following example shows the configuration required for the in-process OracleAS Reports Services server. The lines should be placed within the configuration for the appropriate OC4J instance. The configuration directives are handled by the OC4J module.

```
<ias-component ....>
  <process-type ....>
    <module-data>
      <category id="urlping-parameters">
        <data id="/reports/rwserver/pingserver?start=auto" value="200"/>
      </category>
    </module-data>
    <process-set ....>
    </process-type>
</ias-component>
```

12.4.2 Standalone OracleAS Reports Services Server

The following example shows the configuration required for the standalone OracleAS Reports Services server.

```
<ias-component id="ReportsServer1" status="enabled" id-matching="false">
  <process-type id="ReportsServer1" module-id="ReportsModule">
    <process-set id="ReportsServer1" restart-on-death="true" numprocs="1">
      <environment>
        <variable id ='PATH' value="Put Path_value here" append="true"/>
      </environment>
      <module-data>
        <category id="general-parameters">
          <data id="batch" value="yes"/>
        </category>
        <category id="restart-parameters">
          <data id="reverseping-timeout" value="300"/>
        </category>
      </module-data>
      <start timeout="300" retry="3"/>
      <stop timeout="300"/>
      <restart timeout="300"/>
      <ping timeout="30" interval="30"/>
    </process-set>
  </process-type>
</ias-component>
```

12.5 OracleAS Reports Services Attribute Descriptions

This section describes the attributes that are specific for OracleAS Reports Services.

The OracleAS Reports Services attributes are described with the following format:

- **Title:** This is the attribute name and value being defined. For example, `id="ReportsServer1"`.
- **Required:** This field defines whether or not the attribute is required in the component definition.
- **Default:** This defines the default value assigned to the attribute. The default value appears in the installed version of the `opmn.xml` file or is assigned internally if the attribute is not present.
- **Valid values:** This field defines the valid values for the attribute. For example, `ReportsServer1`.
- **Path:** This field defines in which elements the attribute can appear. For example, `ias-component`.

id="ReportsServer1"

Required: true
Default: none
Valid values: ReportsServer1
Path: ias-component
Path: ias-component/process-type
Path: ias-component/process-type/process-set

This `id` is required and cannot be changed. The name must match the `targets.xml` entry or elements and attributes will not work.

module-id="ReportsModule"

Required: true
 Default: ReportsModule
 Valid values: ReportsModule
 Path: ias-component/process-type

This id defines the type of process and associates the configuration with the OracleAS Reports Services process module.

numprocs="1"

Required: true
 Default: none
 Valid values: 1
 Path: ias-component/process-type/process-set

This attribute gives the number of OracleAS Reports Services servers started for the process-set. Its value has to be 1 because the process-set id identifies the OracleAS Reports Services server.

value="Put Path_value here"

Required: true
 Default: none
 Valid values: The path value has to include the directory containing the uname command.
 Path: ias-component/process-type/process-set/environment/variable

This is the value of the PATH environment variable set for the OracleAS Reports Services process. This variable has to be set for the start script to find "uname".

id="general-parameters"

Required: false
 Default: none
 Path: ias-component/process-type/process-set/module-data/category

This category collects all the data that is common to the various process management tasks.

id="batch"

Required: false
 Default: none
 Valid values: yes, no
 Path: ias-component/process-type/process-set/module-data/category/data

This id is used to construct the command line for starting and stopping of the OracleAS Reports Services server. The information is passed through to OracleAS Reports Services, without interpretation or use by OPMN. If it is not configured, this argument is not passed in.

See Also: Oracle Application Server Reports Services Publishing Reports to the Web

id="restart-parameters"

Required: false

Default: none

Path: ias-component/process-type/process-set/module-data/
category

The `restart-parameters` category for defines parameters used in death detection.

id="reverseping-timeout"

Required: false

Default: 300

Valid values: Any reasonable timeout value specified in seconds.

Path: ias-component/process-type/process-set/module-data/
category/data

If OPMN does not receive a notification within the last `reverseping-timeout` period, it restarts the process.

id="urlping-parameters"

Required: false

Default: none

Path: ias-component/process-type/process-set/module-data/
category

This `id` is a category that collects all the URLs that will be pinged by the OC4J module. The protocol used for pinging is AJPv1.3.

id="/reports/rw servlet/pingserver?start=auto"

Required: false

Default: none

Valid values: /reports/rw servlet/pingserver?start=auto

Path: ias-component/process-type/process-set/module-data/ category/data

This data `id` specifies the URL that will be pinged by the OC4J module. In the context of the in-process OracleAS Reports Services server, pinging this URL allows OPMN to determine if the OracleAS Reports Services server application is responsive or not. If the application is unresponsive, OPMN restarts the corresponding OC4J process.

value="200"

Required: false

Default: none

Valid values: 200

Path: ias-component/process-type/process-set/module-data/ category/data

The `value` specifies a valid HTTP code that is expected in response to the ping request. If the response HTTP code matches the value configured here, OPMN considers the application to be valid. Otherwise, OPMN restarts the OC4J process.

13

Configuring Oracle Application Server Discoverer

This chapter describes Oracle Application Server Discoverer (OracleAS Discoverer) configuration in the OPMN opmn.xml file.

It features the following topics:

- [OracleAS Discoverer Process Module Configuration](#)
- [OracleAS Discoverer Minimum Configuration](#)
- [OracleAS Discoverer Complete Configuration](#)
- [OracleAS Discoverer Attribute Descriptions](#)

13.1 OracleAS Discoverer Process Module Configuration

The following lines load and identify the OracleAS Discoverer process module. Management of OracleAS Discoverer processes by the process module are identified by the module id.

```
<module path="$ORACLE_HOME/opmn/lib/libopmndisco.so" cron="900">
  <module-id id="Disco_OSSAgent"/>
  <module-id id="Disco_OAD"/>
  <module-id id="Disco_PreferenceServer"/>
</module>
```

13.2 OracleAS Discoverer Minimum Configuration

The following lines represent the minimum configuration for OracleAS Discoverer. Default values are assigned to all other configuration elements and attributes for OracleAS Discoverer.

```
<ias-component id="Discoverer">
  <environment>
    <variable id="VBROKER_HOME" value="$ORACLE_HOME/vbroker4"/>
    <variable id="VBROKER_DIR" value="$ORACLE_HOME/vbroker4/bin"/>
    <variable id="OSAGENT_PORT" value="16001"/>
    <variable id="DISCO_DIR" value="$ORACLE_HOME/discoverer904"/>
    <variable id="DISCO_JRE" value="$ORACLE_HOME/jdk/jre"/>
    <variable id="DISCO_PREFERENCE" value="pdsun-dev120oracleDiscovererPreferences902"/>
    <variable id="LD_LIBRARY_PATH" value="$DISCO_DIR/lib:$VBROKER_DIR/lib: $ORACLE_HOME/lib"
      append="true"/>
  </environment>
  <process-type id="OSAgent" module-id="Disco_OSSAgent" working-dir= "$DISCO_DIR/util">
    <process-set id="OSAgent" numprocs="1"/>
  </process-type>
  <process-type id="OAD" module-id="Disco_OAD" working-dir="$DISCO_DIR/util">
    <process-set id="OAD" numprocs="1"/>
  </process-type>
  <process-type id="PreferenceServer" module-id="Disco_PreferenceServer" working-dir= "$DISCO_
DIR/util">
    <process-set id="PreferenceServer" numprocs="1"/>
  </process-type>
</ias-component>
```

13.3 OracleAS Discoverer Complete Configuration

The following lines show a complete configuration for OracleAS Discoverer. It contains all possible configuration elements and attributes for OracleAS Discoverer.

```
<ias-component id="Discoverer">
  <environment>
    <variable id="VBROKER_HOME" value="$ORACLE_HOME/vbroker4"/>
    <variable id="VBROKER_DIR" value="$ORACLE_HOME/vbroker4/bin"/>
    <variable id="OSAGENT_PORT" value="16001"/>
    <variable id="DISCO_DIR" value="$ORACLE_HOME/discoverer904"/>
    <variable id="DISCO_JRE" value="$ORACLE_HOME/jdk/jre"/>
    <variable id="DISCO_PREFERENCE" value="pdsun-dev120oracleDiscoverer Preferences902"/>
    <variable id="LD_LIBRARY_PATH" value="$DISCO_DIR/lib:$VBROKER_DIR/lib:$ORACLE_HOME/lib"
      append="true"/>
  </environment>
  <process-type id="OSAgent" module-id="Disco_OSAgent" working-dir="$DISCO_DIR/util">
    <ping interval="300"/>
  <process-set id="OSAgent" numprocs="1"/>
  <module-data>
    <category id="start-parameters">
      <data id="osagent-options" value="osagent-options_value"/>
      <data id="osagent-extra-options" value="osagent-extra-options"/>
    </category>
    <category id="ping-parameters">
      <data id="osfind-options" value="osfind-options_value"/>
      <data id="osfind-java-options" value="osfind-java-options_value"/>
      <data id="osfind-extra-options" value="osfind-extra-options_value"/>
    </category>
  </module-data>
  </process-type>
  <process-type id="OAD" module-id="Disco_OAD" working-dir="$DISCO_DIR/util">
    <start timeout="180"/>
  <process-set id="OAD" numprocs="1"/>
  <module-data>
    <category id="start-parameters">
      <data id="oad-options" value="oad-options_value"/>
      <data id="oad-java-options" value="oad-java-options_value"/>
      <data id="oad-extra-options" value="oad-extra-options_value"/>
    </category>
    <category id="ping-parameters">
      <data id="oadutil-options" value="oadutil-options_value"/>
      <data id="oadutil-java-options" value="oadutil-java-options_value"/>
      <data id="oadutil-extra-options" value="oadutil-extra-options_value"/>
    </category>
    <category id="sessionserver-parameters">
      <data id="shutdown-event-name" value="shutdown-event-name_value"/>
      <data id="reverse-ping-interval" value="interval-value"/>
    </category>
    <category id="bounce-parameters">
      <data id="bounce-process" value="true"/>
      <data id="bounce-time" value="01:00"/>
    </category>
  </module-data>
  </process-type>
  <process-type id="PreferenceServer" module-id="Disco_PreferenceServer"
  working-dir="$DISCO_DIR/util">
  <process-set id="PreferenceServer" numprocs="1"/>
  <module-data>
    <category id="start-parameters">
      <data id="dis51pr-options" value="dis51pr-options_value"/>
    </category>
  </module-data>

```

```
<data id="dis51pr-extra-options" value="dis51pr-extra-options_value"/>
</category>
</module-data>
</process-type>
</ias-component>
```

13.4 OracleAS Discoverer Attribute Descriptions

This section describes the attributes that are specific for OracleAS Discoverer.

The OracleAS Discoverer attributes are described with the following format:

- **Title:** This is the attribute name and value being defined. For example, `id="Discoverer"`.
- **Required:** This field defines whether or not the attribute is required in the component definition.
- **Default:** This defines the default value assigned to the attribute. The default value appears in the installed version of the `opmn.xml` file or is assigned internally if the attribute is not present.
- **Valid values:** This field defines the valid values for the attribute. For example, `Discoverer`.
- **Path:** This field defines in which elements the attribute can appear. For example, `ias-component`.

id="Discoverer"

Required: true
Default: none
Valid values: Discover
Path: ias-component

This `id` is required and cannot be changed. The name must match the `targets.xml` entry or elements and attributes will not work.

id="VBROKER_HOME"

Required: true
Default: \$ORACLE_HOME/vbroker4
Path: ias-component/environment/variable

OPMN depends on this environment variable to locate the binaries and libraries of vbroker. This element gives you the flexibility to point to the vbroker directory that is different from \$ORACLE_HOME/vbroker4.

id="VBROKER_DIR"

Required: true
Default: \$ORACLE_HOME/vbroker4/bin
Path: ias-component/environment/variable

Visibroker depends on this environment variable to locate the binaries of vbroker.

id="OSAGENT_PORT"

Required: true
 Default: none
 Path: ias-component/environment/variable

This environment variable is required for all the OracleAS Discoverer components to talk to each other.

id="DISCO_DIR"

Required: true
 Default: none
 Path: ias-component/environment/variable

This environment variable is required for OPMN to locate the binaries and libraries of OracleAS Discoverer components, such as dis51pr.

id="DISCO_JRE"

Required: true
 Default: none
 Path: ias-component/environment/variable

This environment variable is used by OPMN to locate the JRE location used to start the Object Activation Daemon (OAD). This element gives you the flexibility to point to the JRE directory other than the one under \$ORACLE_HOME.

id="DISCO_PREFERENCE"

Required: true
 Default: none
 Path: ias-component/environment/variable

This environment variable defines the preference instance name for the preference server.

id="LD_LIBRARY_PATH"

Required: true
 Default: none
 Path: ias-component/environment/variable

This environment variable is required for OracleAS Discoverer binaries to find the libraries.

Three directories are required:

- \$DISCO_DIR/lib
- \$VBROKER_DIR/lib
- \$ORACLE_HOME/lib

id="OSAgent"

Required: true
Default: none
Valid values: OSAgent
Path: ias-component/process-type

This id is required and cannot be changed. The id must match the targets.xml entry or elements and attributes will not work.

module-id="Disco_OSAgent"

Required: true
Default: none
Path: ias-component/process-type

This id defines the type of process and associates this configuration with a process module.

working-dir="\$DISCO_DIR/util"

Required: true
Default: none
Valid values: \$DISCO_DIR/util
Path: ias-component/process-type

This attribute specifies where the log file associated with the osagent process will exist.

interval="300"

Required: false
Default: 20 seconds
Valid values: >= 0
Path: ias-component/process-type/ping

Please refer to the common configuration for a full list of attributes for the ping element. The above example shows how to specify a ping interval to ping your process every 5 minutes.

numprocs="1"

Required: true
Default: none
Valid values: 1
Path: ias-component/process-type/process-set

This attribute gives the number of osagent instances started for this process-set. There should be at most one instance running in one Oracle Application Server instance.

id="start-parameters"

Required: false
Default: none
Path: ias-component/process-type/process-set/module-data/category

This is the category to specify the startup parameters for osagent.

id="osagent-options"

Required: false

Default: Refer to the following paragraph.

Path: ias-component/process-type/process-set/module-data/category/data

The default option for \$ORACLE_HOME/vbroker4/bin/osagent is:

```
"-p $OSAGENT_PORT"
```

You can overwrite this default option by specifying a valid value for this element.

id="osagent-extra-options"

Required: false

Default: Refer to the following paragraph.

Path: ias-component/process-type/process-set/module-data/category/data

You can specify extra options for the default option for "-p \$OSAGENT_PORT" for the osagent-options parameter. If the osagent-options parameter is specified then this id is ignored.

id="ping-parameters"

Required: false

Default: none

Path: ias-component/process-type/process-set/module-data/category

This is the category to specify the ping parameters for osagent.

id="osfind-options"

Required: false

Default: Refer to the following paragraph.

Path: ias-component/process-type/process-set/module-data/category/data

osfind is a Java program. The base class is com.inprise.vbroker.ds.OSFind. The default option is none and the default Java option is:

```
-Xbootclasspath/p:$VBROKER_HOME/lib/vbjorb.jar -Dvbroker.agmt.port=$OSAGENT_PORT"
```

You can overwrite this default option by specifying a valid value for this element.

id="osfind-java-options"

Required: false

Default: Refer to the following paragraph.

Path: ias-component/process-type/process-set/module-data/category/data

You can specify extra Java options to the default Java option of:

```
-Xbootclasspath/p:$VBROKER_HOME/lib/vbjorb.jar -Dvbroker.agmt.port=$OSAGENT_PORT"
```

If osagent-options is specified, this element is ignored.

id="osfind-extra-options"

Required: false

Default: Refer to the following paragraph.

Path: ias-component/process-type/process-set/module-data/category/data

You can specify extra options to the default option of:

-Xbootclasspath/p:\$VBROKER_HOME/lib/vbjorb.jar -Dvbroker.agemt.port=\$OSAGENT_PORT"

If osagent-options is specified, this element is ignored.

id="OAD"

Required: true

Default: none

Valid values: OAD

Path: ias-component/process-type

This name is required and cannot be changed. The name must match the targets.xml entry or elements and attributes will not work.

module-id="Disco_OAD"

Required: true

Default: none

Path: ias-component/process-type

This name defines the type of process and associates this configuration with a process module.

working-dir="\$DISCO_DIR/util"

Required: true

Default: none

Valid values: \$DISCO_DIR/util

Path: ias-component/process-type

Path: ias-component/process-type/process-set

This attribute specifies where the log file associated with the OAD process will exist.

timeout="180"

Required: false

Default: 120 seconds

Valid values: >= 0

Path: ias-component/process-type/start

Please refer to the [Chapter 4, "opmn.xml Common Configuration"](#) for a full list of attributes for the `<start>` element. OAD is a Java process which may take additional time to just start. Use this attribute to adjust it.

numprocs="1"

Required: true
 Default: none
 Valid values: 1
 Path: ias-component/process-type/process-set

This attribute gives the number of OAD instances started for this process-set. There should be at most one OAD instance running in one Oracle Application Server instance.

id="start-parameters"

Required: false
 Default: none
 Path: ias-component/process-type/process-set/module-data/category

This is the category to specify the startup parameters for OAD.

id="oad-options"

Required: false
 Default: *Refer to the following paragraph.*
 Path: ias-component/process-type/process-set/module-data/category/data

oad is a Java program. The base class is com.inprise.vbroker.activationImpl.OADserv. The default option for \$ORACLE_HOME/vbroker4/bin/oad is:

```
-Xbootclasspath:$VBROKER_DIR/lib/vbjorb.jar
-Dvbroker.agent.port=$OSAGENT_PORT
-Dvbroker.se.iiop_tp.scm.iiop_tp.listener.port=0 -DJDKrenameBug=true
-Dvbroker.oad.path=\"\" -Dvbroker.orb.admDir=$VBROKER_ADM
(For Windows only(-Dvbroker.oad.systemRoot=$SystemRoot ))
-Dorg.omg.CORBA.ORBClass=com.inprise.vbroker.orb.ORB
-Dorg.omg.CORBA.ORBSingletonClass=com.inprise.vbroker.orb.ORBSingleton
-cp $VBROKER_DIR/lib/vbjorb.jar
```

The default option is:

```
"-path $VBROKER_ADM"
```

You can overwrite this default option by specify a valid value for this element.

id="oad-java-options"

Required: false
 Default: *Refer to the following paragraph.*
 Path: ias-component/process-type/process-set/module-data/category/data

The default option is:

```
"-path $VBROKER_ADM"
```

You can specify extra Java options to the default option. If oad-options is specified, this element is ignored.

id="oad-extra-options"

Required: false

Default: *Refer to the following paragraph.*

Path: ias-component/process-type/process-set/module-data/
category/data

The default option is:

"-path \$VBROKER_ADMIN"

You can specify extra options to the default option. If oad-options is specified, this element is ignored.

id="ping-parameters"

Required: false

Default: none

Path: ias-component/process-type/process-set/module-data/
category

This is the category to specify the ping parameters for OAD.

id="oadutil-options"

Required: false

Default: *Refer to the following paragraph.*

Path: ias-component/process-type/process-set/module-data/
category/data

oadutil is a Java program. The base class is
`com.inprise.vbroker.activationImpl.OADUtil`. The default Java option is:
-xbootclasspath/p:\$VBROKER_HOME/lib/vbjorb.jar -Dvbroker.agent.port=\$OSAGENT_PORT

The default option is:

"list"

You can overwrite this default option by specify a valid value for this element.

id="oadutil-java-options"

Required: false

Default: *Refer to the following paragraph.*

Path: ias-component/process-type/process-set/module-data/
category/data

The default Java option for is:

-xbootclasspath/p:\$VBROKER_HOME/lib/vbjorb.jar -Dvbroker.agent.port=\$OSAGENT_PORT

The default option is:

"list"

You can specify extra Java options to the default option. If oadutil-options is specified, this element is ignored.

id="oadutil-extra-options"

Required: false

Default: Refer to the following paragraph.

Path: ias-component/process-type/process-set/module-data/category/data

The default Java option for is:

```
-xbootclasspath/p:$VBROKER_HOME/lib/vbjorb.jar -Dvbroker.agent.port=$OSAGENT_PORT
```

The default option is:

"list"

You can specify extra options to the default option. If oadutil-options is specified, this element is ignored.

id="sessionserver-parameters"

Required: false

Default: none

Path: ias-component/process-type/process-set/module-data/category

This is the category to specify the parameters for the session servers started by OAD.

The session server is a OracleAS Discoverer process that is launched during a user session. The OracleAS Discoverer session server performs such operations as connecting to a database or opening a workbook. The session server provides the link between the OracleAS Discoverer servlet or applet to the database.

id="shutdown-event-name"

Required: false

Default: \$ORACLE_HOME/opmn/bin/opmn

Valid values: Any existing file

Path: ias-component/process-type/process-set/module-data/category/data

Shutdown-event-name is used to create either a semaphore on Unix, or an event on Windows in order to coordinate the shutdown of OAD and the session servers. On Unix, it must point to an existing file. If the file does not exist, OPMN will throw an error and disable the whole module.

Oracle does not recommend changing this value.

id="reverse-ping-interval"

Required: false

Default: 300 (in seconds)

Valid values: any value > 0

Path: ias-component/process-type/process-set/module-data/
category/data

Session servers will send reverse ping notifications to OPMN periodically. This id sets the interval of the pinging. The default interval is five minutes. If session server crashes, no more notifications are sent to OPMN. OPMN will remove the entry for this server from the DMS tree if two consecutive notifications are missing from a server. If you modify this value, you should also modify the argument to start the session server accordingly.

id="bounce-parameters"

Required: false

Default: none

Path: ias-component/process-type/process-set/module-data/
category

This is the category to specify the bounce parameters for OAD. OAD can be scheduled for restart every day for a specific time period. It must run for 24 hours prior to schedule set-up.

id="bounce-process"

Required: false

Default: false

Valid values: true or false

Path: ias-component/process-type/process-set/module-data/
category/data

Set this data to true and OAD will be restarted as scheduled.

id="bounce-time"

Required: false

Default: none

Valid values: any value in the format of 00:00 (hour:minute) on a 24-hour base.

Path: ias-component/process-type/process-set/module-data/
category/data

id="PreferenceServer"

Required: true

Default: none

Valid values: PreferenceServer

Path: ias-component/process-type

This name is required and cannot be changed. The name must match the entry in the targets.xml file or elements and attributes will not work.

module-id="Disco_PreferenceServer"

Required: true
 Default: none
 Path: ias-component/process-type

This name defines the type of process and associates the configuration with a process module.

working-dir="\$DISCO_DIR/util"

Required: true
 Default: none
 Valid values: \$DISCO_DIR/util
 Path: ias-component/process-type

This attribute specifies the location of the log file associated with the dis51pr process.

numprocs="1"

Required: true
 Default: none
 Valid values: 1
 Path: ias-component/process-type/process-set

This attribute gives the number of preference server instances started for the process-set. There should be at most one instance running in one Oracle Application Server instance.

id="start-parameters"

Required: false
 Default: none
 Path: ias-component/process-type/process-set/module-data/category

This is the category to specify the startup parameters for dis51pr.

id="dis51pr-options"

Required: false
 Default: *Refer to the following paragraph.*
 Path: ias-component/process-type/process-set/module-data/category/data

The default option for \$DISCO_DIR/bin/dis51pr is:

"-ORBagentPort \$OSAGENT_PORT -preference \$DISCO_PREFERENCE -U"

You can overwrite this default option by specifying a valid value for this element.

id="dis51pr-extra-options"

Required: false

Default: Refer to the following paragraph.

Path: ias-component/process-type/process-set/module-data/
category/data

The default option for \$DISCO_DIR/bin/dis51pr is:

```
"-ORBagentPort $OSAGENT_PORT -preference $DISCO_PREFERENCE -U"
```

You can specify extra options to the listed default option. If dis51pr-options is specified, this element is ignored.

14

Configuring Log Loader

This chapter describes Log Loader configuration in the OPMN `opmn.xml` file.

It features the following topics:

- [Log Loader Process Module Configuration](#)
- [Log Loader Minimum Configuration](#)
- [Log Loader Complete Configuration](#)
- [Log Loader Attribute Descriptions](#)

14.1 Log Loader Process Module Configuration

The following lines load and identify the Log Loader process module. Management of Log Loader processes by the process module are identified by the module id.

```
<module path="%ORACLE_HOME%/opmn/lib/liblogloader.so">
    <module-id id="LOGLDR"/>
</module>
```

14.2 Log Loader Minimum Configuration

The following lines represent the minimum configuration for Log Loader. Default values are assigned to all other configuration elements and attributes for Log Loader.

```
<ias-component id="LogLoader" status="enabled" id-matching="true">
    <process-type id="logloaderd" module-id="LOGLDR" working-dir="$ORACLE_HOME">
        <environment>
            <variable id="CLASSPATH" value="$ORACLE_HOME/diagnostics/lib/ojdl.jar" append="true"/>
            <variable id="CLASSPATH" value="$ORACLE_HOME/lib/xmlparserv2.jar" append="true"/>
            <variable id="CLASSPATH" value="$ORACLE_HOME/jdbc/lib/classes12.jar" append="true"/>
            <variable id="CLASSPATH" value="$ORACLE_HOME/opmn/lib/ons.jar" append="true"/>
            <variable id="CLASSPATH" value="$ORACLE_HOME/opmn/lib/optic.jar" append="true"/>
        </environment>
        <process-set id="logloaderd" restart-on-death="true" numprocs="1">
            <module-data>
                <category id="start-parameters">
                    <data id="java-parameters" value="-server -Doracle.home=$ORACLE_HOME
-Djava.security.policy=$ORACLE_HOME/diagnostics/bin/java.policy"/>
                </category>
                <category id="stop-parameters">
                    <data id="java-parameters" value=" -Doracle.home=$ORACLE_HOME -Djava.security.policy
=$ORACLE_HOME/diagnostics/bin/java.policy"/>
                </category>
            </module-data>
        </process-set>
    </process-type>
</ias-component>
```

14.3 Log Loader Complete Configuration

The following lines show the complete configuration for Log Loader. It contains all possible configuration elements and attributes for Log Loader.

```
<ias-component id="LogLoader" status="enabled" id-matching="true">
  <process-type id="logloaderd" module-id="LOGLDR" working-dir="$ORACLE_HOME">
    <environment>
      <variable id="CLASSPATH" value="$ORACLE_HOME/lib/xmlparserv2.jar" append="true"/>
      <variable id="CLASSPATH" value="$ORACLE_HOME/jdbc/lib/classes12dms.jar" append="true"/>
      <variable id="CLASSPATH" value="$ORACLE_HOME/opmn/lib/ons.jar" append="true"/>
      <variable id="CLASSPATH" value="$ORACLE_HOME/opmn/lib/optic.jar" append="true"/>
      <variable id="CLASSPATH" value="$ORACLE_HOME/dcm/lib/dcm.jar" append="true"/>
      <variable id="CLASSPATH" value="$ORACLE_HOME/diagnostics/lib/ojdl.jar" append="true"/>
    </environment>
    <process-set id="logloaderd" restart-on-death="true" numprocs="1">
      <module-data>
        <category id="start-parameters">
          <data id="java-bin" value="$ORACLE_HOME/jdk/bin/java"/>
          <data id="java-parameters" value="-server-Doracle.home=$ORACLE_HOME
-Djava.security.policy=$ORACLE_HOME/diagnostics/bin/java.policy"/>
          <data id="class-name" value="oracle.core.ojdl.loader.LogLoaderDaemon"/>
          <data id="application-parameters" value="-start -config $ORACLE_HOME/diagnostics/
config/logloader.xml"/>
        </category>
        <category id="stop-parameters">
          <data id="java-bin" value="$ORACLE_HOME/jdk/bin/java"/>
          <data id="java-parameters" value="-Doracle.home=$ORACLE_HOME
-Djava.security.policy=$ORACLE_HOME/diagnostics/bin/java.policy"/>
          <data id="class-name" value="oracle.core.ojdl.loader.LogLoaderDaemon"/>
          <data id="application-parameters" value="-stop"/>
        </category>
      </module-data>
    </process-set>
  </process-type>
</ias-component>
```

14.4 Log Loader Attribute Descriptions

This section describes the attributes that are specific for Log Loader. This section also provides attribute descriptions of the attributes.

The Log Loader attributes are described with the following format:

- **Title:** This is the attribute name and value being defined. For example, `id="LogLoader"`.
- **Required:** This field defines whether or not the attribute is required in the component definition.
- **Default:** This defines the default value assigned to the attribute. The default value appears in the installed version of the `opmn.xml` file or is assigned internally if the attribute is not present.
- **Valid values:** This field defines the valid values for the attribute. For example, `LogLoader`.
- **Path:** This field defines in which elements the attribute can appear. For example, `ias-component`.

id="LogLoader"

Required: true
Default: none
Valid values: LogLoader
Path: ias-component

This id is required and cannot be changed.

id="logloaderd"

Required: true
Default: none
Valid values: logloaderd
Path: ias-component/process-type
Path: ias-component/process-type/process-set

This id is required and cannot be changed.

module-id="LOGLDR"

Required: true
Default: none
Path: ias-component/process-type

This name defines the type of process and associates this configuration with a process module.

numprocs="1"

Required: true
Default: none
Valid values: 1
Path: ias-component/process-type/process-set

This attribute gives the number of LogLoader server instances started for this process-set. The value should always be 1.

id="start-parameters"

Required: true
Default: none
Path: ias-component/process-type/process-set/module-data/
category

This indicates the configuration block for start-parameters category. This category specifies the Java-parameters data must be specified.

id="java-bin"

Required: false
Default: \$ORACLE_HOME/jdk/bin/java
Valid values: Any full file path to a Java executable file
Path: ias-component/process-type/process-set/module-data/
category/data

You can specify alternate paths to the Java executable.

id="java-parameters"

Required: true

Default: None

Valid values: Any valid Java parameters

Path: ias-component/process-type/process-set/module-data/
category/data

The parameters -Doracle.home and -Djava.security.policy are required and must be defined in this section.

id="class-name"

Required: false

Default: oracle.core.ojdl.loader.LogLoaderDaemon

Valid values: a Java class name

Path: ias-component/process-type/process-set/module-data/
category/data

This field allows you to specify a Java class name other than the default to start or stop the LogLoader server process. This id is for internal use and must not be changed by the user.

id="application-parameters"

Required: false

Default: To start: -start -config \$ORACLE_
HOME/diagnostics/config/logloader.xml.

To stop: -stop -config \$ORACLE_
HOME/diagnostics/config/logloader.xml

Valid values: any valid Java application parameters that are recognized by the class specified in "class-name".

Path: ias-component/process-type/process-set/module-data/
category/data

This field allows you to set application parameters. Any parameters defined here will replace the default values.

id="stop-parameters"

Required: false

Default: none

Path: ias-component/process-type/process-set/module-data/
category

This indicates the configuration block for stop-parameters category. In this category, some stop related module data can be specified.

15

Configuring DCM Daemon

This chapter describes DCM Daemon configuration in the OPMN `opmn.xml` file.

It features the following topics:

- [DCM Daemon Process Module Configuration](#)
- [DCM Daemon Minimum Configuration](#)
- [DCM Daemon Complete Configuration](#)
- [DCM Daemon Attribute Descriptions](#)

15.1 DCM Daemon Process Module Configuration

The following lines load and identify the DCM Daemon process module. Management of DCM Daemon processes by the process module are identified by the module id.

```
<module path="$ORACLE_HOME/opmn/lib/libopmnndcmdaemon">
  <module-id id="DCMDaemon" />
</module>
```

15.2 DCM Daemon Minimum Configuration

The following lines represent the minimum configuration for the DCM Daemon. Default values are assigned to all other configuration elements and attributes for DCM Daemon.

```
<ias-component id="dcm-daemon" status="enabled" id-matching="true">
  <process-type id="dcm-daemon" module-id="DCMDaemon">
    <process-set id="dcm" numprocs="1">
      <module-data>
        <category id="start-parameters">
          <data id="jar-file" value="$ORACLE_HOME/dcm/lib/dcm.jar"/>
          <data id="application-parameters" value="daemon -logdir $ORACLE_HOME/dcm/logs/
daemons_logs -o $ORACLE_HOME"/>
        </category>
      </module-data>
    </process-set>
  </process-type>
</ias-component>
```

15.3 DCM Daemon Complete Configuration

The following lines show a complete configuration for the DCM Daemon. It contains all possible configuration elements and attributes for the DCM Daemon.

```
<ias-component id="dcm-daemon" status="enabled" id-matching="true">
  <process-type id="dcm-daemon" module-id="DCMDaemon">
    <process-set id="dcm" numprocs="1">
      <module-data>
        <category id="start-parameters">
          <data id="java-bin" value="$ORACLE_HOME/jdk/bin/java"/>
          <data id="java-parameters" value="-Doracle.ias.sysmgmt.logging.loglevel=ERROR"/>
          <data id="jar-file" value="$ORACLE_HOME/dcm/lib/dcm.jar"/>
          <data id="application-parameters" value="daemon -logdir $ORACLE_HOME/dcm/logs
daemons_logs -o $ORACLE_HOME"/>
        </category>
      </module-data>
    </process-set>
  </process-type>
</ias-component>
```

15.4 DCM Daemon Attribute Descriptions

This section describes the attributes that are specific for the DCM Daemon. This section also provides attribute descriptions of the attributes.

The DCM Daemon attributes are described with the following format:

- **Title:** This is the attribute name and value being defined. For example, `id="dcm-daemon"`.
- **Required:** This field defines whether or not the attribute is required in the component definition.
- **Default:** This defines the default value assigned to the attribute. The default value appears in the installed version of the `opmn.xml` file or is assigned internally if the attribute is not present.
- **Valid values:** This field defines the valid values for the attribute. For example, `dcm-daemon`.
- **Path:** This field defines in which elements the attribute can appear. For example, `ias-component`.

id="dcm-daemon"

Required: true
 Default: none
 Valid values: `dcm-daemon`
 Path: `ias-component`
 Path: `ias-component/process-type`

This `id` is required and cannot be changed.

module-id="DCMDaemon"

Required: true
 Default: none
 Path: `ias-component/process-type`

This `id` defines the type of process and associates this configuration with a process module. This name is required and cannot be changed.

numprocs="1"

Required: true
 Default: none
 Valid values: 1
 Path: `ias-component/process-type/process-set`

This attribute gives the number of DCM Daemon instances started for the `process-set`. The value should always be 1.

id="start-parameters"

Required: true
 Default: none
 Path: `ias-component/process-type/process-set/module-data/category`

This indicates the configuration block for the `start-parameters` category. In this category, at a minimum, the Java-parameters data must be specified.

id="java-bin"

Required: false
Default: \$ORACLE_HOME/jdk/bin/java
Valid values: Full path to java.exe
Path: ias-component/process-type/process-set/module-data/category/data

You can specify alternate paths to the Java Interpreter (executable).

id="java-parameters"

Required: false
Default: -Doracle.ias.sysmgmt.logging.loglevel=ERROR
Valid values: Any valid Java parameters
Path: ias-component/process-type/process-set/module-data/category/data

This field allows you to specify any valid Java parameters. All specified Java parameters overwrite the default values. If the loglevel is not specified, the loglevel for the DCM daemon is set at the ERROR level.

id="jar-file"

Required: true
Default: \$ORACLE_HOME/dcm/lib/dcm.jar
Valid values: full path to valid dcm.jar
Path: ias-component/process-type/process-set/module-data/category

This field specifies a JAR file for starting and stopping the DCM daemon. This id is required and cannot be changed.

id="application-parameters"

Required: true
Default: daemon -logdir \$ORACLE_HOME/dcm/logs/daemon_logs -o \$ORACLE_HOME
Valid values: any valid Java application parameters that are recognized by the main-class specified in dcm.jars manifest.mf
Path: ias-component/process-type/process-set/module-data/category/data

This field allows you to set application parameters. Parameters defined here replace the default values. In order to start the DCM Daemon, it is mandatory to supply daemon, -logdir, and -o parameters.

16

Configuring Custom Process

This chapter describes custom process configuration in the OPMN opmn.xml file.

It features the following topics:

- [Custom Process Module Configuration](#)
- [Custom Process Minimum Configuration](#)
- [Custom Process Complete Configuration](#)
- [Custom Process Attribute Descriptions](#)

16.1 Custom Process Module Configuration

The following lines load and identify the custom process module. Management of custom processes by the process module are identified by the module id.

```
<module path="%ORACLE_HOME%/opmn/lib/libopmncustom.so">
  <module-id id="CUSTOM" />
</module>
```

16.2 Custom Process Minimum Configuration

The following lines represent the minimum configuration for a custom process. Default values are assigned to all other configuration elements and attributes for the custom process.

```
<ias-component id="Custom">
  <process-type id="Custom" module-id="CUSTOM">
    <process-set id="Custom" numprocs="1">
      <module-data>
        <category id="start-parameters">
          <data id="start-executable" value="Your start executable here" />
        </category>
      </module-data>
    </process-set>
  </process-type>
</ias-component>
```

16.3 Custom Process Complete Configuration

[Example 16-1](#) show a complete configuration for a custom process. It contains all possible configuration elements and attributes for a custom process.

A custom process can be part of any other Oracle Application Server component. In such cases, the `process-type` element in [Example 16-1](#) must be part of the component configuration.

16.3.1 Ping

The custom module provides the framework for pinging a custom process in one of two ways:

- HTTP ping
- script ping

The type of ping can be configured by specifying the appropriate data in the `ping-parameters` category. The sample Oracle Application Server configuration example [Example 16-1](#) shows a custom process using HTTP ping. [Example 16-2](#) is an example of script ping that you can substitute into the component configuration.

Example 16–1 Custom Process Complete Configuration

```

<ias-component id="Custom" status="enabled" id-matching="false">
    <environment>
        <variable id="TEST_ENV_VARIABLE" value="/your/test/value"
            append="false"/>
    </environment>
    <process-type id="Custom" module-id="CUSTOM">
        <process-set id="Custom" restart-on-death="true" numprocs="1">
            <module-data>
                <category id="start-parameters">
                    <data id="start-executable" value="Your start executable here" />
                    <data id="start-args" value="Your start args here" />
                </category>
                <category id="stop-parameters">
                    <data id="stop-executable" value="Your stop executable here" />
                    <data id="stop-args" value="Your stop args here" />
                </category>
                <category id="restart-parameters">
                    <data id="restart-executable" value="Your restart executable here" />
                    <data id="restart-args" value="Your restart args here" />
                </category>
                <category id="ping-parameters">
                    <data id="ping-type" value="http" />
                    <data id="ping-url" value="/your/ping/url" />
                    <data id="ping-host" value="abc.company.com" />
                    <data id="ping-port" value="7777" />
                    <data id="ping-limit" value="3" />
                    <data id="ping-timeout" value="300" />
                </category>
                <category id="ready-parameters">
                    <data id="use-ping-for-ready" value="false" />
                </category>
            </module-data>
        </process-set>
    </process-type>
</ias-component>

```

Pinging with a script can be configured as shown in [Example 16–2](#)

Example 16–2 Ping Type Script

```

<category id="ping-parameters">
    <data id="ping-type" value="script" />
    <data id="script-executable" value="Ping executable here" />
    <data id="script-args" value="Ping arguments here " />
</category>

```

You can use pinging (when available) for determining the readiness of a process. This indicates that OPMN needs confirmation that a managed process has started successfully after creation. Processes can inform OPMN of their ready status in various ways. The custom module allows these processes to communicate readiness via pinging. If you configure pinging for a custom process, you can also use this mechanism to determine if the process is ready. You can choose not to configure any mechanism for determining readiness in which case the custom module just assumes that the process started successfully.

Note: The ready ping, if configured, is created soon after the process is created. If the process takes a while to initialize and respond to pings, then using pinging for determining readiness is not appropriate. This is because if the process does not respond to the "ready ping", OPMN will determine that the process did not start correctly and stop it.

16.4 Custom Process Attribute Descriptions

This section describes the attributes that are specific for a custom process. This section also provides attribute descriptions of the attributes.

The custom process attributes are described with the following format:

- **Title:** This is the attribute name and value being defined. For example, `id="Custom"`.
- **Required:** This field defines whether or not the attribute is required in the component definition.
- **Default:** This defines the default value assigned to the attribute. The default value appears in the installed version of the `opmn.xml` file or is assigned internally if the attribute is not present.
- **Valid values:** This field defines the valid values for the attribute. For example, `custom`.
- **Path:** This field defines in which elements the attribute can appear. For example, `ias-component`.

id="Custom"

Required: true

Default: none

Valid values: Any `id` of your choice

Path: `ias-component`

Path: `ias-component/process-type`

Path: `ias-component/process-type/process-set`

This `id` is required and can be any name you choose. The `id` cannot be a duplicate of existing names.

module-id="CUSTOM"

Required: true

Default: none

Valid values: The same as the `module-id` specified in [Section 16.1, "Custom Process Module Configuration"](#).

Path: `ias-component/process-type`

The `module-id` associates the process with a module. For Custom processes, this `id` has to match the `module-id` specified in Process Module Configuration for the Custom module.

id="start-parameters"

Required: true

Default: none

Path: ias-component/process-type/process-set/module-data/category

The start-parameters category contains child elements specifying the start executable and start arguments.

id="start-executable"

Required: true

Default: none

Valid values: a valid executable to run

Path:

ias-component/process-type/process-set/module-data/category/data

This data element specifies the name of the executable to be started.

id="start-args"

Required: false

Default: none

Valid values: Valid arguments to the executable specified by start-executable data element.

Path:

ias-component/process-type/process-set/module-data/category/data

The value of this data element should be a string containing all the arguments for the start executable. Multiple data elements with this id should not be specified.

id="stop-parameters"

Required: false

Default: none

Path: ias-component/process-type/process-set/module-data/category

The stop-parameters category contains child elements specifying the stop executable and stop arguments. If this category is not configured, OPMN stops the process with the kill command.

id="stop-executable"

Required: false

Default: none

Path:

ias-component/process-type/process-set/module-data/category/data

This data element specifies the name of the executable to be used for stopping the process.

id="stop-args"

Required: false

Default: none

Path:

ias-component/process-type/process-set/module-data/category/
data

The value of this data element should be a string containing all the arguments to the stop executable. Multiple data elements with this id should not be specified.

id="restart-parameters"

Required: false

Default: none

Path: ias-component/process-type/process-set/module-data/
category

The restart-parameters category contains child elements specifying the restart executable and restart arguments. This category needs to be configured if the process has an explicit restart command. In the absence of a restart command, a stop followed by the start command will be executed whenever the process needs to be restarted.

When restart data is specified, OPMN assumes that the process ID of the process remains the same after a restart. If there is no explicit restart command available for the process, a stop followed by a start is issued. In this scenario, a process ID change is acceptable.

id="restart-executable"

Required: false

Default: none

Valid values: A valid restart executable name

Path:

ias-component/process-type/process-set/module-data/category/
data

This data element specifies the name of the executable to be used for restarting the process.

id="restart-args"

Required: false

Default: none

Valid values: valid arguments to the restart executable

Path:

ias-component/process-type/process-set/module-data/category/
data

The value of this data element should be a string containing all the arguments to the restart executable. Multiple data elements with this id should not be specified.

id="ping-parameters"

Required: false

Default: none

Path: ias-component/process-type/process-set/module-data/category

Custom processes that are pinged via the HTTP protocol must specify this category. This module data category consists of all the data required to perform such a ping.

id="ping-type"

Required: false

Default: none

Valid values: http, script

Path:

ias-component/process-type/process-set/module-data/category/
data

Custom processes that wish to be pinged have to specify this module data.

Note Also:

- [Example 16-1, "Custom Process Complete Configuration"](#)
- [Example 16-2, "Ping Type Script"](#)

id="ping-url"

Required: false

Default: /

Valid values: Any valid URL

Path:

ias-component/process-type/process-set/module-data/category/
data

This data element is used to specify the URL at which the process will be pinged. The listed parameters are used for HTTP pings.

id="ping-host"

Required: false

Default: none

Valid values: A valid hostname to which a custom process is bound.

Path:

ias-component/process-type/process-set/module-data/category/
data

This data element is used to specify the host name to which a custom process is bound. If this data is not specified, pinging will not be performed. If an invalid hostname is specified, the process-set will be disabled.

id="ping-port"

Required: false

Default: none

Valid values: A valid port at which a custom process is listening for HTTP requests

Path:

ias-component/process-type/process-set/module-data/category/
data

The port at which a custom process is listening. If this data is not specified, pinging will not be performed. If an invalid port is specified, the process-set will be disabled.

id="ping-limit"

Required: false

Default: 3

Valid values: Any reasonable value that reflects the tolerance that OPMN should have for failed pings. This tolerance is used by OPMN to determine when the process should be declared unresponsive and restarted.

Path: ias-component/process-type/process-set/module-data/
category/data

This module data element defines the tolerance for failed pings. After the number of ping failures reaches this limit, the process is deemed unresponsive and restarted by OPMN.

id="ping-timeout"

Required: false

Default: 300 seconds

Valid values: Any reasonable timeout value

Path:

ias-component/process-type/process-set/module-data/category/
data

The timeout value specified with this data element is used as the maximum time OPMN will wait for a ping response. If a response is not obtained within the timeout period, the ping attempt will be considered a failure.

id="script-executable"

Required: false

Default: none

Valid values: A valid script executable

Path: ias-component/process-type/process-set/module-data/
category/data

This data element specifies the name of the executable to be used for pinging the process. An exit value of 0 from this executable is considered success. All other values indicate a ping failure.

id="script-args"

Required: false
Default: none
Valid values: valid arguments to the ping executable
Path: ias-component/process-type/process-set/module-data/category/data

The value of this data element should be a string containing all the arguments to the ping executable. Multiple data elements with this id should not be specified.

id="ready-parameters"

Required: false
Default: none
Path: ias-component/process-type/process-set/module-data/category

The module data category to indicate if pinging should be used to determine that a custom process started successfully.

id="use-ping-for-ready"

Required: false
Default: false
Valid values: true or false
Path: ias-component/process-type/process-set/module-data/category/data

The value of this data element determines if pinging should be used to determine if a process is available.

17

OPMN Troubleshooting

This chapter describes some troubleshooting tips for OPMN. It features the following topics:

- [Diagnosing OPMN Problems](#)
- [Common Problems](#)

17.1 Diagnosing OPMN Problems

There are several methods for troubleshooting any problems you may have using OPMN:

- [OPMN log Files](#)
- [opmnctl debug](#)
- [iHAT](#)
- [Oracle Enterprise Manager 10g Application Server Control Console](#)
- [Troubleshooting with Event Scripts](#)

17.1.1 OPMN log Files

The OPMN log files enable you to troubleshoot difficulties you might have in execution and use of OPMN and OPMN-managed processes. OPMN and OPMN-managed processes generate log files during processing. You can review the following generated log files to verify successful or unsuccessful execution of an OPMN command:

- *ORACLE_HOME/opmn/logs*: The standard output and standard error of OPMN managed processes. OPMN creates a log file for each component and assigns a unique concatenation of the Oracle Application Server component with a number. For example, the standard output log for OracleAS Web Cache may be *WebCache~WebCacheAdmin~1*. The process specific console logs are the first and best resource for investigating problems related to starting and stopping components.
- *ORACLE_HOME/opmn/logs/ipm.log*: Review the error codes and messages that are shown in the *ipm.log* file. OPMN generates and outputs the error messages in this file. The *ipm.log* file tracks command execution and operation progress. The level of detail that gets logged in the *ipm.log* can be modified by configuration in the *opmn.xml* file. Refer to [Chapter 4, "opmn.xml Common Configuration"](#) for examples of debug levels.
- *ORACLE_HOME/opmn/logs/ons.log*: Use the *ons.log* file to debug the ONS portion of OPMN or for early OPMN errors. The ONS portion of OPMN is initialized before PM, and so errors that occur early in OPMN initialization will show up in the *ons.log* file.
- *ORACLE_HOME/opmn/logs/service.log*: (Microsoft Windows only). The *service.log* displays any error messages generated by OPMN services while interacting with service control manager.

17.1.2 opmnctl debug

Use the *opmnctl debug* command to verify the status of an Oracle Application Server process and whether any actions are pending. This command generates output that can be used in conjunction with contact to your local Oracle support to diagnose your OPMN problem.

The syntax for the *opmnctl debug* command is:

```
opmnctl [<scope>] debug [comp=pm|ons] [interval=<secs> count=<num>]
```

where *@scope* is the optional scope for the request.

Output is generated following execution of the opmnctl debug command. Oracle recommends that you contact Oracle support to use the generated output to assist in diagnosis of your problem.

The attributes (<attr>) name for this command are either comp, interval, or count . The value for comp can be either ons or pm, representing ONS and PM, respectively. If comp is not specified, then both ons and pm debug information is reported. For example, the following command outputs debug information for ONS.

```
prompt > opmnctl debug comp=ons
```

You can specify the interval in seconds and number of requests sent to OPMN to assist in the debugging process. The values of <interval> and <count> must always be specified together. Values for them should be integers greater than 0. For example, the following command, outputs debug information at an interval of 5 seconds 3 times.

```
prompt > opmnctl debug comp=pm interval=5 count=3
```

Contact your local Oracle support to assist you in using the opmnctl debug command to diagnose your OPMN problem.

17.1.3 iHAT

Use the iHAT tool to provide a real time, graphical interface view of your enterprise. iHAT displays all Oracle Application Server processes managed by one or more OPMN servers including useful performance metrics about each process as obtained from DMS. The snapshot of the system is updated continuously at a configurable interval.

To download iHAT, visit the Oracle Technology Network (OTN):

<http://otn.oracle.com/membership>

17.1.4 Oracle Enterprise Manager 10g Application Server Control Console

Application Server Control Console provides a graphical interface that enables diagnosis of Oracle Application Server components in your network and enterprise. Application Server Control Console features a log page. The log page enables you to view all of the Oracle Application Server log files in one place and trace problems across multiple log files. Application Server Control Console uses an API that contacts OPMN.

You can use Application Server Control Console to enable or disable Oracle Application Server components: You can disable components so they do not start when you start an Oracle Application Server instance.

See Also: *Oracle Application Server 10g Administrator's Guide*

17.1.5 Troubleshooting with Event Scripts

You can create your own event scripts that record Oracle Application Server process event activities. You can create a script that records events prior to the start or stop of Oracle Application Server processes, as well as an unscheduled system crash.

Refer to the `<event-scripts>` element description in [Chapter 4, "opmn.xml Common Configuration"](#).

The following [Example 17–1](#) shows a pre-start event script:

Example 17–1 Pre-start Event Script

```
#!/bin/sh
echo
echo =====
echo ===== PRE-START EVENT SCRIPT =====
echo =====

timeStamp="N/A"
instanceName="N/A"
componentId="N/A"
processType="N/A"
processSet="N/A"
processIndex="N/A"
stderrPath="N/A" # not available w/pre-start unless part of restart
stdoutPath="N/A" # not available w/pre-start unless part of restart
reason="N/A"
pid="N/A" # only available with pre-stop, post-crash
startTime="N/A" # only available with pre-stop, post-crash

while [ $# -gt 0 ]; do
    case $1 in
        -timeStamp)   timeStamp=$2; shift;;
        -instanceName) instanceName=$2; shift;;
        -componentId) componentId=$2; shift;;
        -processType) processType=$2; shift;;
        -processSet) processSet=$2; shift;;
        -processIndex) processIndex=$2; shift;;
        -stderr) stderrPath=$2; shift;;
        -stdout) stdoutPath=$2; shift;;
        -reason) reason=$2; shift;;
        -pid) pid=$2; shift;;
        -startTime) startTime=$2; shift;;
        *) echo "Option Not Recognized: [$1]"; shift;;
    esac
    shift
done

echo timeStamp=$timeStamp
echo instanceName=$instanceName
echo componentId=$componentId
echo processType=$processType
echo processSet=$processSet
echo processIndex=$processIndex
echo stderr=$stderrPath
echo stdout=$stdoutPath
echo reason=$reason
echo pid=$pid
echo startTime=$startTime
```

Note: The pre-start event script example, [Example 17-1](#), will not work for the Microsoft Windows operating system; however, you can create a script, with a .bat suffix, with similar functionality.

Use the full path to the .bat file when adding the necessary configuration information to the opmn.xml file,

17.2 Common Problems

This section describes some of the common problems encountered when using OPMN. It features the following topics:

- [Oracle Application Server Process Does Not Start](#)
- [Determining if Oracle Application Server Processes are Dying or Unresponsive](#)
- [opmnctl Command Execution Times Out](#)
- [Oracle Application Server Component Automatically Restarted by OPMN](#)
- [opmn.xml Environment Variables](#)
- [Unexpected opmnctl start Behavior](#)
- [Disabled Element in the opmn.xml File](#)
- [Management of Oracle9iAS Release 2 \(9.0.2 and 9.0.3\) Instances](#)

17.2.1 Oracle Application Server Process Does Not Start

If you are unable to start an Oracle Application Server process using OPMN:

- Verify and if necessary, correct, the command input. Confirm the spelling and choice of option for the command you are entering.

Note: Do not use command line scripts or utilities from previous versions of Oracle9iAS for starting OPMN or Oracle Application Server components.

- Review the standard out output log for the Oracle Application Server process. Output from the process console is located in the *ORACLE_HOME/opmn/logs* directory. For example, the standard output log for OracleAS Web Cache may be *WebCache~WebCacheAdmin~1*.
- Verify the dependency requirements for the Oracle Application Server process you are attempting to start. For example, the Oracle Application Server process you want to start may require that the Oracle Application Server Metadata Repository be up and running.
- Verify the element values for the Oracle Application Server component in the opmn.xml file. Use the opmnctl validate command to verify configuration of opmn.xml file. You may have mis-configured the opmn.xml for the Oracle Application Server component you are attempting to start.

17.2.2 Determining if Oracle Application Server Processes are Dying or Unresponsive

If your Oracle Application Server processes are dying or unreachable:

- Look at the *ORACLE_HOME/opmn/logs/ipm.log* for Oracle Application Server processes at level of 4 or higher. Look for process crashed or process unreachable messages. OPMN automatically restarts Oracle Application Server processes that die or become unresponsive.
- Create event scripts for any pre-stop or post-crash events. The event scripts could be used to create a specific log file or send you an email about a failure.

See Also: [Section 17.1.5, "Troubleshooting with Event Scripts"](#)

- Review the Oracle Application Server component specific output in the *ORACLE_HOME/opmn/logs*.
- Review the Oracle Application Server component specific log file located in Oracle Application Server component directory.

See Also: [Section 17.1.1, "OPMN log Files"](#)

- Use iHAT to view the actions of Oracle Application Server

See Also: [Section 17.1.3, "iHAT"](#)

17.2.3 opmnctl Command Execution Times Out

The time it takes to execute an *opmnctl* command is dependent on the type of Oracle Application Server process and available computer hardware. Because of this the time it takes to execute an *opmnctl* command may not be readily apparent.

The default start time out for OC4J is approximately five minutes. If an OC4J process is does not start-up after an *opmnctl* command, OPMN will wait approximately an hour before timing out and aborting the request.

To verify successful execution of the *opmnctl* command, try the following:

1. Increase the *start* element *timeout* attribute for the component that is not starting. Set the *timeout* in the *opmn.xml* file at a level that will allow OPMN to wait for process to come up. This functionality is available with the *startproc* command which will start all the relevant processes configured in *opmn.xml*.
2. Check the *start* element in the *opmn.xml* file and change the *retry* attribute to a higher increment of time.
3. Look at the *ORACLE_HOME/opmn/logs/* for the Oracle Application Server process that is not starting.
4. Review the component-specific log file for the Oracle Application Server component that is not starting. For example, *ORACLE_HOME/discoverer/logs*.
5. Examine the *ORACLE_HOME/opmn/logs/ipm.log* for any indication of problems. Increase the log level in the file to obtain additional information.

See Also: [Chapter 4, "opmn.xml Common Configuration"](#)

6. Contact Oracle Technical Support.

If you are having difficulty with an Oracle Application Server instance that is part of a farm, review the `ons.log` file for the Oracle Application Server instance. The "attempting active connection init" message in the `ons.log` file indicates that there is another OPMN configured in the farm that is currently shut-down. OPMN tries to consistently connect to the shut-down OPMN. Use the listed steps to determine why the down OPMN is not running.

17.2.4 Oracle Application Server Component Automatically Restarted by OPMN

If an Oracle Application Server component is automatically restarted by OPMN, try the following:

- Review the message for the Oracle Application Server component in the `ipm.log` file.
- Verify that the ping timeout for the Oracle Application Server component is sufficient. An Oracle Application Server component that receives a lot of activity may require an increase in the length of time for the timeout. Increase the ping timeout element in the Oracle Application Server component `opmn.xml` file.

17.2.5 `opmn.xml` Environment Variables

The environment variable used to launch OPMN server is not inherited by the Oracle Application Server process started by OPMN server. OPMN sets the environment variables at the `ias-instance` level, with the values extracted either from the `ias-instance` configuration or from the OPMN run time environment.

See Also: [Chapter 4, "opmn.xml Common Configuration"](#)

17.2.6 Unexpected `opmnctl start` Behavior

Occasionally, there is unexpected behavior when you use the `opmnctl start` command to start OPMN; either only OPMN is started or OPMN makes a best effort to start Oracle Application Server OPMN-managed processes. Typically, this unexpected behavior is due to turning-off or rebooting your computer without first shutting down OPMN. When you restart your computer, all OPMN-managed processes are started.

Oracle recommends that you shutdown OPMN before shutting down your computer. Use the `opmnctl stopall` command to stop OPMN and OPMN-managed processes.

On the Microsoft Windows operating system, you can use the Windows services control panel to stop OPMN and OPMN-managed processes.

17.2.7 Disabled Element in the `opmn.xml` File

If you are unable to start an Oracle Application Server process, check if an element in the Oracle Application Server `opmn.xml` file is disabled. If an element in the `opmn.xml` file is disabled OPMN will generate an output message of "Missing" or "Disabled".

17.2.8 Management of Oracle9iAS Release 2 (9.0.2 and 9.0.3) Instances

By default, the SSL element in the 10g (9.0.4) opmn.xml file is enabled; however, the SSL element is not enabled in the opmn.xml file for Oracle9iAS Release 2 (9.0.2 and 9.0.3).

If you have an Oracle Application Server farm containing Oracle9iAS Release 2 (9.0.2 and 9.0.3) instances you must enable the SSL element in the Release 2 opmn.xml file. Refer to the *Oracle9i Application Server Administrator's Guide*, Release 2 (9.0.2) for the steps to enable the SSL element.

Index

A

adapter-framework, 11-7
af-class-name id, 11-7
af-string id, 11-6
ajp id, 6-3, 9-3
ajp ports, 9-3
alert | datafeeder | performance | messaging | notification
n module-id, 10-7
Application Server Control, 2-2, 2-5, 2-9, 17-3
application-parameters id, 15-4
async, 2-12, 2-15
attribute
 syntax, 2-14
attribute name, 2-12

B

batch id, 12-5
bounce-parameters id, 13-12
bounce-process id, 13-12
bounce-time id, 13-12

C

cache-timeout attribute, 4-19
category element, 4-12
child process, 2-23
clu, 2-17
cluster
 syntax, 2-13
cmp, 2-17
command, 2-8
command definitions
 opmnctl, 2-13
complete configuration
 custom process, 16-2
 DCM Daemon, 15-2
 Log Loader, 14-3
 OC4J, 6-2
 Oracle HTTP Server, 5-2
 Oracle Internet Directory, 8-3
 OracleAS Discoverer, 13-3
 OracleAS Port Tunnel, 9-2
 OracleAS ProcessConnect, 11-3
 OracleAS Reports Services, 12-3

OracleAS Web Cache, 7-2
OracleAS Wireless, 10-3
components, OPMN-managed, 1-2
config-file id, 5-4, 6-5
configset id, 8-7
connect id, 8-5
count, 17-3
cpu, 2-17
custom
 process module, 16-2
custom id, 16-4
custom module
 ping, 16-2
CUSTOM module-id, 16-4
custom process
 complete configuration, 16-2
 minimum configuration, 16-2

D

data element, 4-12
database element, 4-17
db-connect-info element, 8-5
DCM, 1-6
DCM Daemon
 complete configuration, 15-2
 minimum configuration, 15-2
 process module, 15-2
dcm-daemon id, 15-3
DCMDaemon module-id, 15-3
default start time, 17-6
dependencies element, 4-16
dis51pr-extra-options id, 13-14
dis51pr-options id, 13-13
DISCO_DIR id, 13-5
DISCO_DIR/util
 working-dir attribute, 13-6
DISCO_JRE id, 13-5
Disco_OAD module-id, 13-8
Disco_OSAgent module-id, 13-6
DISCO_PREFERENCE id, 13-5
Disco_PreferenceServer module-id, 13-13
Discoverer id, 13-4
dms id, 11-7
DMS port value, 11-5
dms-port-property-name id, 11-5

E

environment element, 4-14
event script
 post-crash, 1-4
 pre-start, 1-4
 pre-stop, 1-4
event script arguments, 4-24
event scripts, 1-4, 17-4
event-scripts element, 4-24
example elements, 4-2
explicit, OPMN-managed components, 1-2

F

farm
 syntax, 2-14
-fmt
 option, 2-16
-fmtlist
 syntax, 2-17
fmtlist, 2-16
form factor key, 2-22
.formfactor file., 2-22
-fsep
 option, 2-16

G

general-parameters id, 12-5

H

help
 opmnctl, 2-8, 2-19
HTTP ping, 16-2
HTTP_Server id, 5-3

I

ias-component, 2-11, 2-12
ias-component element, 4-15
ias-instance attribute, 4-21
ias-instance element, 4-13
IASPT id, 9-3
IASPT module-id, 9-3
idx, 2-17
iHAT, 2-23, 17-3
im-class-name id, 11-6
implicit, OPMN-managed components, 1-2
im-string id, 11-6
in-process mode
 OracleAS Reports Services, 12-2, 12-3
ins, 2-17
instance
 syntax, 2-13
instance value, 8-6
integration-manager|, 11-7
interval, 13-6, 17-3
ipaddr element, 4-4
ipm.log, 17-2, 17-6, 17-7

ip-parameters id, 11-4

J

jar-file id, 15-4
java options id, 6-4
java-bin id, 6-5, 10-8
justification, 2-17

L

-1
 option, 2-16
LD_LIBRARY_PATH id, 13-5
local instance
 starting, 3-2
 stopping, 3-2
local port, 2-23
log files
 OPMN, 17-2
Log Loader
 complete configuration, 14-3
 minimum configuration, 14-2
 process module, 14-2
log pag, 17-3
log-dir, 11-5
log-file element, 4-5, 4-8
LOGLDR module-id, 14-4
LogLoader id, 14-4
logloaderd id, 14-4

M

managed-process element, 4-21
max_retry, 2-19
mem, 2-17
minimum configuration
 custom process, 16-2
 DCM Daemon, 15-2
 Log Loader, 14-2
 OC4J, 6-2
 Oracle HTTP Server, 5-2
 Oracle Internet Directory, 8-2
 OracleAS Discoverer, 13-2
 OracleAS Port Tunnel, 9-2
 OracleAS ProcessConnect, 11-2
 OracleAS Reports Services, 12-2
 OracleAS Web Cache, 7-2
 OracleAS Wireless, 10-2
mode, 2-12
module, 4-9
module element, 4-9
module-data element, 4-11
module-id element, 4-13
modules
 PM, 1-5, 2-4
multiple instance
 starting, 3-3
 stopping, 3-3

N

-noheaders
 option, 2-16
no-reverseping-failed-ping-limit id, 5-5, 6-6
notification-server element, 4-4
numprocs, 5-3
numprocs attribute, 5-3, 7-4
numprocs element, 5-3

O

OAD id, 13-8
oad-extra-options, 13-10
oad-java-options, 13-9
oadutil-extra-options id, 13-11
oadutil-java-options id, 13-10
OC4J
 complete configuration, 6-2
 minimum configuration, 6-2
 process module, 6-2
OC4J element, 6-3
OC4J id, 6-3
OC4J module-id, 6-3
oc4j-options id, 6-5
OHS module id, 5-3
OID element, 4-18
OID id, 8-4
OID module-id, 8-4
oidmon-parameters id, 8-5
ONS, 4-4
ONS (Oracle Notification Server), 2-3
ons.log, 17-2, 17-7
OPMN, 1-2, 1-5, 2-2, 2-3, 2-5
 functionality, 2-2
 log files, 17-2
OPMN daemon, 2-9, 2-12
opmn element, 4-4
OPMN local listener, 2-22
opmnctl, 1-3, 2-2, 2-9
 command definitions, 2-13
 help, 2-8, 2-19
 options, 2-16
 ping, 2-8, 2-19
 process control commands, 2-11
 quick reference, 2-8
 reload, 2-8, 2-10, 2-11
 restartproc, 2-8, 2-11
 shutdown, 2-8, 2-10
 start, 2-8, 2-9
 startall, 2-8, 2-10
 startproc, 2-8, 2-11
 status, 2-8, 2-18
 stopall, 2-8, 2-10
 stopproc, 2-8, 2-10, 2-11
 syntax, 2-7
 usage, 2-8, 2-10, 2-20
 validate, 2-8, 2-21
 verbose, 2-15
opmnctl attributes, 2-14
opmnctl commands, 2-8

opmnctl debug, 17-2
opmnctl usage status, 2-18
opmnctl validate, 17-5
opmn.xml file, 1-5, 2-4, 2-6
option
 -fmt, 2-16
 -fsep, 2-16
 -l, 2-16
 -noheaders, 2-16
 -rsep, 2-16
options, 2-8
 opmnctl, 2-16
Oracle Enterprise Manager Application Server Control, 2-5
Oracle HTTP Server
 complete configuration, 5-2
 minimum configuration, 5-2
 process module, 5-2
Oracle Internet Directory
 complete configuration, 8-3
 minimum configuration, 8-2
 process module, 8-2
Oracle Notification Server see ONS
Oracle Process Manager Modules see PM Modules, 2-4
Oracle Process Manager see PM, 2-4
Oracle wallet, 4-7
OracleAS Discoverer
 complete configuration, 13-3
 minimum configuration, 13-2
 process module, 13-2
OracleAS Port Tunnel
 complete configuration, 9-2
 minimum configuration, 9-2
 process module, 9-2
OracleAS ProcessConnect
 complete configuration, 11-3
 minimum configuration, 11-2
OracleAS Reports Services, 12-2
 complete configuration, 12-3
 in-process mode, 12-2, 12-3
 process module, 12-2
 standalone mode, 12-2, 12-3, 12-4
OracleAS Web Cache
 complete configuration, 7-2
 minimum configuration, 7-2
 process module, 7-2
OracleAS Wireless
 complete configuration, 10-3
 minimum configuration, 10-2
 process module, 10-2
OracleASProcessConnect
 process module, 11-2
OSAgent id, 13-6
OSAGENT_PORT id, 13-5
osagent-extra-options, 13-7
osagent-options id, 13-7
osfind-java-options, 13-7, 13-8
osfind-options id, 13-7
OSSO element, 4-20

P

path attribute, 4-6
pid, 2-17
ping
 custom module, 16-2
 opmnctl, 2-8, 2-19
ping element, 4-27
ping parameters category, 5-4
ping timeout, 17-7
ping-host id, 16-7
ping-limit id, 16-8
ping-parameters id, 13-7, 13-10, 16-7
ping-port id, 16-8
ping-timeout id, 16-8
ping-type id, 16-7
ping-url id, 5-5, 16-7
PM, 2-4
 modules, 1-5
PM Modules, 2-4
por, 2-17
port element, 4-5, 4-28
post-crash
 event script, 1-4
post-crash element, 4-26
PreferenceServer id, 13-12
pre-start
 event script, 1-4
pre-stop
 event script, 1-4
pre-stop element, 4-25
process control command
 opmnctl, 2-11
process module
 custom, 16-2
 DCM Daemon, 15-2
 Log Loader, 14-2
 OC4J, 6-2
 Oracle HTTP Server, 5-2
 Oracle Internet Directory, 8-2
 OracleAS Port Tunnel, 9-2
 OracleAS ProcessConnect, 11-2
 OracleAS Reports Services, 12-2
 OracleAS Web Cache, 7-2
 OracleAS Wireless, 10-2
ProcessConnect id, 11-4
process-manager element, 4-7
process-modules element, 4-9
process-set, 2-11, 2-12
process-set element, 4-29
process-set id, 11-8
process-set-id attribute, 4-22
process-type, 2-11, 2-12
process-type element, 4-23
process-type id attribute, 4-22
prs, 2-17
prt, 2-17
publish-subscribe model, 2-3
Put Path_value here value, 12-5

Q

quick reference
 opmnctl, 2-8

R

range, 11-7
range attribute, 6-4
range value, 9-3
ready-parameters id, 16-9
reload
 opmnctl, 2-8, 2-11
remote instance
 starting, 3-2
 stopping, 3-2
remote port, 2-23
remote request, 2-23
ReportsModule module-id, 12-5
/reports/rwervlet/pingserver?start=auto id, 12-6
ReportsServer1 id, 12-4
request validation, 2-22
restart element, 4-27
restart timeout, 10-9
restart-args id, 16-6
restart-executable id, 16-6
restart-on-death attribute, 7-4
restart-parameters category, 5-5
restart-parameters id, 6-5
restartproc
 opmnctl, 2-8, 2-11
reverseping-failed-ping-limit element, 6-6
reverseping-failed-ping-limit id, 5-6, 6-6
reverse-ping-interval id, 13-12
reverseping-timeout id, 5-5, 6-6
rmi id, 6-4
-rsep
 option, 2-16
run-oidctl id, 8-6

S

scope, 1-3, 2-7, 2-13
script ping, 16-2
script-args id, 16-9
script-executable id, 16-8
sessionserver-parameters id, 13-11
shutdown
 opmnctl, 2-8, 2-10
shutdown-command id, 11-6
shutdown-event-name id, 13-11
sleep id, 8-6
SSL, 2-22
ssl, 4-18
SSL element, 17-8
ssl element, 4-7, 4-20
sta, 2-17
standalone mode
 OracleAS Reports Services, 12-2, 12-3, 12-4
standard out output log, 17-5
start

opmnctl, 2-8, 2-9
start element, 4-26, 17-6
start order dependencies, 1-4
startall
 opmnctl, 2-8, 2-10
start-args id, 16-5
start-command id, 11-6
start-executable id, 16-5
starting
 local instance, 3-2
 multiple instance, 3-3
 remote instance, 3-2
starting OPMN, 3-2
startodisrv id, 8-8
startoidldapd id, 8-7
startoidrepld id, 8-7
start-parameters category, 5-4
start-parameters id, 6-4
startproc
 opmnctl, 2-8, 2-11
statname, 2-17
status
 opmnctl, 2-8, 2-18
stm, 2-17
stop element, 4-27
stop timeout, 10-9
stopall
 opmnctl, 2-8, 2-10
stop-args id, 16-6
stop-executable id, 16-5
stop-parameters id, 6-5
stopping
 local instance, 3-2
 multiple instance, 3-3
 remote instance, 3-2
stopproc
 opmnctl, 2-8, 2-10, 2-11
sync, 2-12, 2-15
Syntax
 opmnctl ping, 2-19
syntax, 2-14
 attribute, 2-14
 cluster, 2-13
 farm, 2-14
 -fmtlist, 2-17
 instance, 2-13
 opmnctl, 2-7
 opmnctl help, 2-19
 opmnctl options, 2-16
 opmnctl reload, 2-11
 opmnctl restartproc, 2-11
 opmnctl shutdown, 2-10
 opmnctl start, 2-9
 opmnctl startall, 2-10
 opmnctl startproc, 2-11
 opmnctl status, 2-18
 opmnctl stopall, 2-10
 opmnctl stopproc, 2-11
 opmnctl usage, 2-20
 opmnctl validate, 2-21

opmnctl verbose, 2-15
scope, 2-13
value, 2-14

T

targets.xml file, 9-3
timeout, 2-12, 2-15
timeout attribute, 4-19, 17-6
timeout value, 2-10
tnsnames.ora file, 8-5
typ, 2-17

U

ui, 2-17
uid, 2-16
uniqueid, 2-12, 2-16
urlping-parameters id, 12-6
usage
 opmnctl, 2-8, 2-10, 2-20
use-ping-for-ready id, 16-9
utm, 2-17

V

validate
 opmnctl, 2-8, 2-21
value, 2-14
 syntax, 2-14
variable element, 4-14
VBROKER_DIR id, 13-4
VBROKER_HOME id, 13-4
verbose, 2-7
 opmnctl, 2-15

W

wallet file, 2-23
WebCache id, 7-3
WebCache module-id, 7-3
WebCacheAdmin id, 7-3
width, 2-17
wireless id, 10-6
working-dir attribute, 4-24, 8-4
 DISCO_DIR/util, 13-6

