Veritas™ High Availability Agent for Oracle e-Business Components Installation and Configuration Guide

AIX, HP-UX, Linux, Solaris

5.0

✶symantec.™

Veritas High Availability Agent for Oracle e-Business Components Installation and Configuration Guide

The software described in this book is furnished under a license agreement and may be used only in accordance with the terms of the agreement.

Agent version: 5.0.2.0

Document version: 5.0.2

Legal Notice

Technical Support

Symantec Technical Support maintains support centers globally. Technical Support's primary role is to respond to specific queries about product features and functionality. The Technical Support group also creates content for our online Knowledge Base. The Technical Support group works collaboratively with the other functional areas within Symantec to answer your questions in a timely fashion. For example, the Technical Support group works with Product Engineering and Symantec Security Response to provide alerting services and virus definition updates.

Symantec's maintenance offerings include the following:

- A range of support options that give you the flexibility to select the right amount of service for any size organization
- Telephone and Web-based support that provides rapid response and up-to-the-minute information
- Upgrade assurance that delivers automatic software upgrade protection
- Global support that is available 24 hours a day, 7 days a week
- Advanced features, including Account Management Services

For information about Symantec's Maintenance Programs, you can visit our Web site at the following URL:

www.symantec.com/techsupp/

Contacting Technical Support

Customers with a current maintenance agreement may access Technical Support information at the following URL:

www.symantec.com/business/support/assistance_care.jsp

Before contacting Technical Support, make sure you have satisfied the system requirements that are listed in your product documentation. Also, you should be at the computer on which the problem occurred, in case it is necessary to replicate the problem.

When you contact Technical Support, please have the following information available:

- Product release level
- Hardware information
- Available memory, disk space, and NIC information
- Operating system

- Version and patch level
- Network topology
- Router, gateway, and IP address information
- Problem description:
  - Error messages and log files
  - Troubleshooting that was performed before contacting Symantec
  - Recent software configuration changes and network changes

Licensing and registration

If your Symantec product requires registration or a license key, access our technical support Web page at the following URL:

www.symantec.com/techsupp/

Customer service

Customer service information is available at the following URL:

www.symantec.com/techsupp/

Customer Service is available to assist with the following types of issues:

- Questions regarding product licensing or serialization
- Product registration updates, such as address or name changes
- General product information (features, language availability, local dealers)
- Latest information about product updates and upgrades
- Information about upgrade assurance and maintenance contracts
- Information about the Symantec Buying Programs
- Advice about Symantec's technical support options
- Nontechnical presales questions
- Issues that are related to CD-ROMs or manuals

Documentation feedback

Your feedback on product documentation is important to us. Send suggestions for improvements and reports on errors or omissions to clustering_docs@symantec.com. Include the title and document version (located on the second page), and chapter and section titles of the text on which you are reporting.

Maintenance agreement resources

If you want to contact Symantec regarding an existing maintenance agreement, please contact the maintenance agreement administration team for your region as follows:

Asia-Pacific and Japan	contractsadmin@symantec.com
Europe, Middle-East, and Africa	semea@symantec.com
North America and Latin America	supportsolutions@symantec.com

Additional enterprise services

Symantec offers a comprehensive set of services that allow you to maximize your investment in Symantec products and to develop your knowledge, expertise, and global insight, which enable you to manage your business risks proactively.

Enterprise services that are available include the following:

Symantec Early Warning Solutions	These solutions provide early warning of cyber attacks, comprehensive threat analysis, and countermeasures to prevent attacks before they occur.
Managed Security Services	These services remove the burden of managing and monitoring security devices and events, ensuring rapid response to real threats.
Consulting Services	Symantec Consulting Services provide on-site technical expertise from Symantec and its trusted partners. Symantec Consulting Services offer a variety of prepackaged and customizable options that include assessment, design, implementation, monitoring, and management capabilities. Each is focused on establishing and maintaining the integrity and availability of your IT resources.
Educational Services	Educational Services provide a full array of technical training, security education, security certification, and awareness communication programs.

To access more information about Enterprise services, please visit our Web site at the following URL:

www.symantec.com

Select your country or language from the site index.

Contents

Introducing the Veritas High Availability Agent for Oracle e-Business Components

This chapter includes the following topics:

About the Veritas agent for Oracle e-Business Components

The Veritas High Availability agents monitor specific resources within an enterprise application. They determine the status of resources and start or stop them according to external events.

The Veritas agent for Oracle e-Business Components provides high availability for all Oracle e-Business components in a cluster.

See the following Technical Support TechNote for the latest updates or software issues for this agent:

http://seer.entsupport.symantec.com/docs/282004.htm

The agent for Oracle e-Business Components brings the component instances online, monitors the instances, and brings the instances offline. The agent monitors the system processes and server states, and can shutdown the Oracle e-Business component instance in case of a failover. It supports both, 11i and R12 releases of Oracle e-Business Component.

The agent manages the following Oracle e-Business components:

Version	Components
R12	■ Administration Listener ■ OPMN ■ OC4J Forms Server ■ OC4J OACORE ■ OC4J OAFM ■ HTTP Server ■ Fulfillment Server
11i	■ Administration Listener ■ Forms Server ■ Forms Metrics Server ■ Forms Metrics Client ■ Forms Listener ■ FulfillmentServer ■ Report Server ■ Discoverer 4 ■ 9iAS R1 Web Application Server ■ 9iAS R1 Web Application Server Listener

Note: To configure Discoverer 10g component, you must install Oracle AS9 agent.

Supported software

The Veritas agent for Oracle e-Business Components supports the following software versions:

Veritas Cluster Server	■ AIX—VCS 4.0, 5.0 ■ HP-UX—VCS 4.1, 5.0 ■ Linux—VCS 4.0, 4.1, 5.0 ■ Solaris—VCS 4.0, 4.1, 5.0
ACC Library	5.1 and later

Operating Systems	■ AIX 5.3, OSlevel 5
	■ HP-UX 11i v2
	■ Red Hat Enterprise Linux 3.0, 4.0
	■ Solaris 8, 9, 10 on SPARC
Oracle e-Business Suite	11i (11.5.7, 11.5.8, 11.5.9, and 11.5.10)
	R12 (12 and 12.0.4)

Oracle e-Business Components agent functions

The agent consists of resource type declarations and agent executables. The agent executables are organized into online, offline, monitor, and clean functions.

The following sections elaborate the steps that each agent function performs.

Online

The online function performs the following tasks:

■ Performs a preliminary check to ensure that the Oracle e-Business component instance is not running already on the specified node in the cluster.

■ If any Oracle e-Business component processes are running, the function performs a clean operation to end these processes.

■ Depending upon the component, the online function begins an Oracle supplied startup script to start the instance.

■ For the specific component instance, the online function waits for the respective processes to start successfully.

Review the startup scripts for each component and the list of processes that the script starts for a component instance.

See "Detecting application failure" on page 13.

The online function exits either after all the processes start successfully, or after the timeout period specified in the OnlineTimeout attribute expires.

Offline

The offline function performs the following tasks:

■ Performs a preliminary check to ensure that the Oracle e-Business component instance is not already offline on the specified node in the cluster.

■ Depending upon the component, the offline function begins an Oracle supplied stop script to stop the instance.

■ For the specific component instance, the offline function waits for the respective processes to stop successfully.

Review the startup scripts for each component and the list of processes that the script stops for a component instance.

See "Detecting application failure" on page 13.

The offline function exits either after all the processes stop successfully, or after the timeout period specified in the OfflineTimeout attribute expires.

Monitor

The monitor function monitors the state of the Oracle e-Business component instance on all nodes in the cluster. The function performs the following tasks:

■ Conducts a first level check on the specific component instance to ensure that the processes are running smoothly.

■ If the SecondLevelMonitor attribute is greater than zero, the monitor function performs a thorough check of all the components running.
Review the information about how first and second level monitor functions work.
See "Detecting application failure" on page 13.

■ Depending upon the MonitorProgram attribute, the monitor function can perform a customized check using a user supplied monitoring utility.
Review the details about executing a custom monitor program.
See "Executing a customized monitoring program" on page 29.

Clean

The clean function performs the following tasks:

■ Checks the reason why the clean function was invoked.

■ If the function was invoked because of a failed offline operation, the clean function looks for all the processes running for the specified component instance, and cleans the processes.

■ In all other cases, the clean function attempts to gracefully shutdown the specified component instance.

If a graceful shutdown fails, the clean function looks for all the processes running for the specified component instance, and cleans the processes.

Detecting application failure

The agent monitor function is responsible for checking if a particular Oracle e-Business component instance is running successfully. The function performs a two-level process check to verify that the processes are running for an instance.

The function performs a primary process check to verify the existence of appropriate component processes. The function uses the ServerType attribute value to determine which processes must be present for a particular Oracle e-Business component.

Table 1-1 lists the scripts and processes for Oracle e-Business 11i component instances.

Table 1-1   Scripts and Processes for Oracle e-Business 11i Component Instances

Oracle component	Oracle script to start or stop the component	List of processes
Forms Server	adfrmctl.sh	f60srvm, f60webmx
Forms Metric Server	adfmsctl.sh	d2ls60
Forms Metric Client	adfmcctl.sh	d2lc60
Reports Server	adrepctl.sh	rwmts60
Web Server	adapcctl.sh	httpd
Discoverer	addisctl.sh	osagent, oad
RPC Listener	adalnctl.sh	tnslsnr
Fulfillment Server	jtffmctl.sh	java.*FulfillmentServer*

Table 1-2 lists the scripts and process for Oracle e-Business R12 component instances

Table 1-2   Scripts and Processes for Oracle e-Business R12 Component Instances

Oracle component	Oracle script to start or stop the component	List of processes
OPMN	adopmnctl.sh	opmn
Forms Server (Servlet Mode)	adformsctl.sh	forms
Forms Server (Socket Mode)	adformsrvctl.sh	frmsrv

Table 1-2 Scripts and Processes for Oracle e-Business R12 Component Instances *(continued)*

Oracle component	Oracle script to start or stop the component	List of processes
OACORE Server	adcorectl.sh	oacore
OAFM Server	adoafmctl.sh	oafm
Web Server	adapcctl.sh	httpd
RPC Listener	adalnctl.sh	tnslsnr
Fulfillment Server	jtffmctl.sh	java.*FulfillmentServer*

If the value of the SecondLevelMonitor attribute is greater than zero, the monitor function proceeds to perform deeper and more thorough check on the instance processes.

The following checks are performed for 11i Oracle e-Business Components:

■ If ServerType is equal to Listener, the monitor function performs the TNSPING Test. This test uses the Oracle supplied TNSPING command utility to verify that an application listener is responding to the SQLNET requests.

■ For all other values of the ServerType attribute, the monitor function performs a socket connection test on the port that is specified for the particular component. If the socket connection is functional, the monitor function reports the instance as online.

The following checks are performed for R12 Oracle e-Business Components:

■ If ServerType is equal to Listener, the monitor function performs the LSNRCTL Test. This test uses the Oracle supplied LSNRCTL command utility to verify that an application listener is responding to the SQLNET requests.

■ If the ServerType is opmn, the monitor function performs `opmnctl status` test, where opmnctl is a command provided by oAS 10g to verify the opmn and managed processes.

■ For all other ServerType values, the monitor function runs the `opmnctl status -noheaders -fmt prt%sta` command and filters the ServerType.

For example, for the processes in instance: VIS_ebiz1.ebiz1.vxindia.veritas.com, the output of the command `$ opmnctl status -noheaders -fmt prt%sta` shows:

```
oafm | Alive
forms | Alive
```

```
oacore | Alive
HTTP_Server | Alive
```

Note: OPMN does not manage Form Server, if configured in Socket Mode. In this case Second Level monitor performs socket connection test on the port on which the Form Server is running.

Installing, upgrading, and removing the agent for Oracle e-Business Components

This chapter includes the following topics:

- Before you install the Veritas agent for Oracle e-Business Components
- Installing the ACC library
- Installing the agent in a VCS environment
- Removing the agent in a VCS environment
- Removing the ACC library
- Upgrading the agent for Oracle e-Business Components

Before you install the Veritas agent for Oracle e-Business Components

You must install the Veritas agent for Oracle e-Business Components on all the systems that will host a Oracle e-Business Components service group.

Ensure that you meet the following prerequisites to install the agent for Oracle e-Business Components.

- Install and configure Veritas Cluster Server.

- Remove any previous version of this agent.

- Install the latest version of ACC Library.
  To install or update the ACC Library package, locate the library and related documentation on the agentpack disc.
  See "About ACC Library" on page 18.

Prerequisites for installing the agent to support Solaris zones

Ensure that you meet the following prerequisites to install the agent for Oracle e-Business Components:

- Install Oracle e-Business Components inside the Solaris zones.

- Install and configure the VCS 5.0 environment to support Solaris zones. Refer to the VCS user documentation for details.

- Install the required version of ACC Library.

- Remove any previous version of this agent.

About ACC Library

The operations for the Veritas agent for Oracle e-Business Components depend on a set of Perl modules known as the ACC library. The library must be installed on each system in the cluster that will run the agent for Oracle e-Business Components. The ACC library contains common, reusable functions that perform tasks, such as process identification, logging, and system calls.

Installing the ACC library

Install the ACC library on each system in the cluster that runs an agent that depends on the ACC library.

To install the ACC library

1 Log in as superuser.

2 Navigate to the pkgs directory (the pkgs directory on the CD).

AIX	*cd_mount*/aix/application/acc_library/vcs/*version*_library/pkgs
HP-UX	*cd_mount*/hpux/generic/application/acc_library/vcs/*version*_library/pkgs
Linux	*cd_mount*/linux/generic/application/acc_library/vcs/*version*_library/rpms
Solaris	*cd_mount*/solaris/*dist_arch*/application/acc_library/vcs/*version*_library/pkgs
	where *dist_arch* is sparc or sol_x64.

3 Install the package. Enter **Yes** if asked to confirm overwriting of files in the existing package.

AIX	# installp -ac -d VRTSacclib.rte.bff VRTSacclib.rte
HP-UX	# swinstall -s `pwd` VRTSacclib
Linux	# rpm -i \ VRTSacclib-*VersionNumber*-GA_GENERIC.noarch.rpm
Solaris	# pkgadd -d . VRTSacclib

4 For HP-UX, install the HP-UX patch PHCO_29042 if it is not already installed.

Installing the agent in a VCS environment

Install the agent for Oracle e-Business Components on each node in the cluster.

To install the agent

1 Log in as superuser.

2 Navigate to the directory containing the package for the platform running
in your environment.

AIX   *cd_mount*/aix/application/oracleapps_agent/

vcs_*version*/*version*_agent/pkgs

HP-UX   *cd_mount*/hpux/generic/application/oracleapps_agent/

vcs_*version*/*version*_agent/pkgs

Linux   *cd_mount*/linux/generic/
application/oracleapps_agent/vcs_*version*/
*version*_agent/rpms

Solaris   *cd_mount*/solaris/*dist_arch*/application/

oracleapps_agent/vcs_*version*/*version*_agent/pkgs

3 Install the package.

AIX   `# installp -ac -d VRTSvcsoa11.rte.bff VRTSvcsoa11.rte`

HP-UX   `# swinstall -s 'pwd' VRTSvcsoa11`

Linux   `# rpm -ihv \`
`VRTSvcsoa11-`*AgentVersion*`-GA_GENERIC.noarch.rpm`

Solaris   `# pkgadd -d . VRTSvcsoa11`

Removing the agent in a VCS environment

You must uninstall the agent for Oracle e-Business Components from a cluster
while the cluster is active.

To uninstall the agent in a VCS environment

1   Log in as a superuser.

2   Set the cluster configuration mode to read/write by typing the following command from any node in the cluster:

```
# haconf -makerw
```

3   Remove all Oracle e-Business Components resources from the cluster. Use the following command to verify that all resources have been removed:

```
# hares -list Type=OracleApps
```

4   Remove the agent type from the cluster configuration by typing the following command from any node in the cluster:

```
# hatype -delete OracleApps
```

Removing the agent's type file from the cluster removes the include statement for the agent from the main.cf file, but the agent's type file is not removed from the cluster configuration directory. You can remove the agent's type file later from the cluster configuration directory.

5   Save these changes. Then set the cluster configuration mode to read-only by typing the following command from any node in the cluster:

```
# haconf -dump -makero
```

6   Use the platform's native software management program to remove the agent for Oracle e-Business Components from each node in the cluster.

Execute the following command to uninstall the agent:

AIX	`# installp -u VRTSvcsoa11.rte`
HP-UX	`# swremove VRTSvcsoa11`
Linux	`# rpm -e VRTSvcsoa11`
Solaris	`# pkgrm VRTSvcsoa11`

Removing the ACC library

Perform the following steps to remove the ACC library.

To remove the ACC library

1 Ensure that all agents that use ACC library are removed.

2 Run the following command to remove the ACC library package.

AIX	`# installp -u VRTSacclib.rte`
HP-UX	`# swremove VRTSacclib`
Linux	`# rpm -e VRTSacclib`
Solaris	`# pkgrm VRTSacclib`

Upgrading the agent for Oracle e-Business Components

To upgrade the agent, first remove the older version of the agent.

See "Removing the agent in a VCS environment" on page 20.

Then, follow the instructions to install the new agent software.

See "Installing the agent in a VCS environment" on page 19.

Preparing to configure the agent for Oracle e-Business Components

This chapter includes the following topics:

About configuring the Veritas agent for Oracle e-Business Components

After installing the Veritas agent for Oracle e-Business Components, you must import the agent type configuration file. After importing this file, you can create and configure a Oracle e-Business Components resource. Before you configure a resource, review the attributes table that describes the resource type and its attributes.

To view the sample agent type definition and service groups configuration.

See "About sample configurations for the agent for Oracle e-Business Components" on page 45.

Importing the agent types files for VCS

To use the agent for Oracle e-Business Components, you must import the agent types file into the cluster.

To import the agent types file using the Veritas Cluster Server graphical user interface

1   Start the Veritas Cluster Manager and connect to the cluster on which the agent is installed.

2   Click **File > Import Types**.

3   In the Import Types dialog box, select the following file:

VCS 4.x	/etc/VRTSvcs/conf/sample_OracleApps/OracleAppsTypes.cf
VCS 5.0	/etc/VRTSagents/ha/conf/OracleApps/OracleAppsTypes.cf
VCS 4.x with Solaris zone support	/etc/VRTSvcs/conf/sample_OracleApps/OracleAppsTypes_zones.cf
VCS 5.0 with Solaris zone support	/etc/VRTSagents/ha/conf/OracleApps/OracleAppsTypes_zones.cf

4   Click **Import**.

5   Save the VCS configuration.

The Oracle e-Business Components agent type is now imported to the VCS engine.

You can now create Oracle e-Business Components resources. For additional information about using the VCS GUI, refer to the *Veritas Cluster Server User's Guide*.

Agent attributes for Oracle e-Business components

Table 3-1 lists the attributes required for configuring an Oracle component instance.

Table 3-1   Required attributes

Required attributes	Description
ResLogLevel	The logging detail performed by the agent for the resource. Valid values are:
	ERROR: Only logs error messages.
	WARN : Logs above plus warning messages.
	INFO: Logs above plus informational messages.
	TRACE: Logs above plus trace messages. TRACE is very verbose and should only be used during initial configuration or for troubleshooting and diagnostic operations.
	Type and dimension: string-scalar
	Default: INFO
	Example: TRACE
User	UNIX user name used to start and stop an Oracle e-Business component instance. The clean operation also uses this user name to kill residual processes.
	Type and dimension: string-scalar
	Default: ""
	Example: oracle
OracleAppsVersion	The Oracle Applications e-Business version number
	The valid values are:
	■ 11- For configuring Oracle Applications e-Business 11i release
	■ 12- For configuring Oracle Applications e-Business R12 release
	Type and dimension: integer
	Default: ""
	Example: 12

Table 3-1   Required attributes *(continued)*

Required attributes	Description
ORACLE_HOME	The absolute path to the directory that contains the Oracle client libraries, that are located within the Oracle e-Business software directory.
	Note: The R12 release follows the absolute path as 10.1.2 ORACLE_HOME for Listener and 10.1.3 ORACLE_HOME for OPMN and its managed processes and Fulfillment Server.
	This attribute is used in the monitor function to locate the environment setup file, and the Oracle client binaries that are used in second-level monitor routines.
	For 11i release, this attribute is not required:
	■ when the SecondLevelMonitor attribute is equal to 0. ■ when the ServerType attribute is equal to FormsMetricsServer, FormsMetricsClient, FormsServer, ReportServer, or WebServer.
	Type and dimension: string-scalar
	Default: ""
	Example for 11i: /ebiz/visora/8.0.6
	Example for R12 (for 10.1.2 ORACLE_HOME): /ebiz/oracle/VIS/apps/tech_st/10.1.2
	Example for R12 (for 10.1.3 ORACLE_HOME): /ebiz/oracle/VIS/apps/tech_st/10.1.3
ScriptHome	The absolute path of the Oracle e-Business scripts directory. This directory contains the scripts to start and stop an e-Business component instance.
	Type and dimension: string-scalar
	Default: ""
	Example for 11i: $COMMON_TOP/admin/scripts/VIS_ebiz
	Example for R12: $INST_TOP/admin/scripts

Table 3-1   Required attributes *(continued)*

Required attributes	Description
EnvFile	The directory path of the file that must be sourced with the UNIX shell.
	■ For Oracle Apps 11i, you must source this file to set the environment before executing agent scripts for online, offline, monitor, and clean functions.
	■ For Oracle Apps R12, you must source this file only for SecondLevelMonitoring
	Type and dimension: string-scalar
	Default: ""
	Example for 11i: /ebiz/visappl/APPSVIS_ebiz.env
	Example for R12: /ebiz/oracle/inst/apps/VIS_ebiz1/ora/10.1.3/VIS_ebiz.env
ServerType	Type of server.
	Valid values for Oracle Apps 11i are:
	■ Listener
	■ FormsServer
	■ FormsMetricsServer
	■ FormsMetricsClient
	■ ReportServer
	■ WebServer
	■ Discoverer
	■ FulfillmentServer
	Valid values for Oracle Apps R12 are:
	■ OPMN
	■ WebServer
	■ OAFM
	■ OACORE
	■ FormsServer
	■ Listener
	■ FulfillmentServer
	Type and dimension: string-scalar
	Default: ""
	Example: FormsServer

Table 3-1   Required attributes *(continued)*

Required attributes	Description
Port	The port number that is associated with a particular type of server, as specified in the ServerType attribute.
	The monitor function uses this attribute as an argument for second-level monitor check.
	For Oracle Apps 11i, this attribute is not required when the,
	■ SecondLevelMonitor attribute is equal to 0. ■ ServerType attribute is equal to FormsMetricsServer, FormsMetricsClient, Discoverer.
	Note: For Oracle Apps R12 this attribute is reqd only when the FormServer is configured in socket mode.
	Type and dimension: integer-scalar
	Default: ""
	Example: 8080
DBConnectString	The string used to connect database to the listener. This attribute is used while performing second-level monitoring on the listener.
	Type and dimension: string-scalar
	Default: ""
	Example: VIS
VirtualHostname	The virtual hostname that is associated with an Oracle e-Business component instance.
	Type and dimension: string-scalar
	Default: ""
	Example: ora_com_01

Table 3-2 lists the optional attributes for configuring an Oracle component instance.

Table 3-2   Optional attributes

Optional attributes	Description
MonitorProgram	The full pathname and command-line arguments for an externally provided monitor program.
	For information about setting this attribute, refer to See "Executing a customized monitoring program" on page 29.
	Type and dimension: string-scalar
	Default: ""
	Example 1: /usr/ebiz/visappl/work/myMonitor.sh
	Example 2: /usr/ebiz/visappl/work/myMonitor.sh arg1 arg2
SecondLevelMonitor	Used to enable second-level monitoring. Second-level monitoring is a deeper, more thorough state check of the configured Oracle Apps components. The numeric value specifies how often the monitoring routines must run. 0 means never run the second-level monitoring routines, 1 means run routines every monitor interval, 2 means run routines every second monitor interval, and so on.
	Note: Exercise caution while setting SecondLevelMonitor to large numbers. For example, if the MonitorInterval is set to 100, then the agent executes a process check every 100 minutes, which may not be as often as intended. For maximum flexibility, no upper limit is defined for SecondLevelMonitor.
	Type and dimension: integer-scalar
	Default: 0
	Example: 1

Executing a customized monitoring program

The monitor function executes a custom monitor program to perform a user-defined Oracle e-Business Components instance server state check.

■ The MonitorProgram attribute value is set to a valid executable program.

■ The first level process check indicates that the Oracle e-Business Components instance is online.

■ The SecondLevelMonitor attribute is either set to 0 (false), or SecondLevelMonitor is set to 1 (true) and the second level check indicates that the Oracle e-Business Components instance is online.

This feature allows cluster administrators to define custom programs that can further determine the state of the Oracle e-Business Components instance.

The monitor function interprets the utility exit code as follows:

110 or 0	Oracle e-Business Components server instance is online
100 or 1	Oracle e-Business Components server instance is offline
99	Oracle e-Business Components server instance is unknown
Any other value	Oracle e-Business Components server instance is unknown

To ensure that the custom monitor program is always available to the agent application, Symantec recommends storing the file in a shared directory that is available on an online Oracle system.

Creating soft links to JDBC properties file

Each Oracle application component attempts to connect to the Oracle database for configuration information, for example, instance metadata. For a successful database session, the application accesses the JDBC properties file to obtain the correct database session properties' information. The JDBC file name must be VirtualName_DBNAME, where VirtualName is the virtual name of the Oracle application database resource and DBNAME is the name of the database that is specified during installation.

To avoid problems while accessing the JDBC properties file, Symantec recommends creating a soft link to the JDBC properties file for each node in the cluster.

For example, consider a cluster that has four nodes, oranode1, oranode2, oranode3, and oranode4. The virtual name of the Oracle application database resource is ORADB. The DBNAME is vis. The resource is an Oracle 9iAS instance, where $APPL_TOP is equal to /ora_apps/web/visappl. The 9iAS UNIX user ID is oraweb.

To create a soft link for each node of the cluster, login as oraweb user and run the following commands:

```
$ cd /ora_apps/web/visappl/fnd/11.5.0/secure
$ ln -s  ORADB_vis.dbc oranode1_vis.dbc
$ ln -s  ORADB_vis.dbc oranode2_vis.dbc
$ ln -s  ORADB_vis.dbc oranode3_vis.dbc
$ ln -s  ORADB_vis.dbc oranode4_vis.dbc
```

If you configure a Concurrent Manager instance in a clustered environment, you must create a soft link to the virtual hostname of the CM instance. For example,

if the virtual hostname running the CM instance is oraccms, run this command to create a soft link:

```
$ ln -s  ORADB_vis.dbc oraccms_vis.dbc
```

Note: Oracle Apps R12 has inbuilt and separate instance homes for each instance. Additionally, the DB_TOP, APPL_TOP and COMMON_TOP instances are shared by all separate instances. Thus, in case of Oracle Apps R12, you are not required to create the soft links to JDBC properties file.

Configuring Oracle e-Business Components resources for Solaris zones support

To enable the agent for Oracle e-Business Components to support Solaris zones, ensure that you perform the following configuration steps:

- Install Oracle e-Business Components on dedicated Solaris zones.

- Preferrably, follow the Symantec recommendation of installing zones on a shared disk for convenient configuration, failover, and maintenance.

- Make sure that the name of the Solaris zone is the same as the virtual host name that you use to install and configure the Oracle e-Business Components.

Configuring the service groups for Oracle e-Business Components

This chapter includes the following topics:

- Installing Oracle e-Business components for clustering purposes

- Configuring Oracle e-Business components for cluster support

Installing Oracle e-Business components for clustering purposes

Perform the following steps to install the Oracle e-Business components and prepare the nodes for clustering.

Decide the type of installation as single-node or multiple-node

You can either install all the e-Business components on a single node in a cluster, or you can install the components on multiple nodes in the cluster.

Assume that you install the e-Business components on multiple nodes, as shown in the following example

For Oracle Apps 11i:

- Install the Oracle database component on node 1.

- Install the Oracle Web Server component on node 2.

- Install the Forms Server and the Reports Server on node 3.

For Oracle Apps R12:

- Install Web and Forms services on node 1.

- Install Applications database services on node 2.

- Install Concurrent processing services on node 3.

Configure a disk group and a file system

Depending upon the number of components to install, configure a disk group and file system that is large enough to hold the components.

Refer to the Oracle e-Business installation documentation for details about space requirements for each e-Business component.

If you decide to create a VISION demo database, allocate space in the disk group and the file system accordingly.

Obtain virtual IP addresses for each node

For each Oracle e-Business component, you must configure a virtual IP address. Configure the virtual IP addresses, and resolve the addresses by adding them to the local files or to the DNS system.

Install the Oracle e-Business component

For each component perform the following steps:

To install the Oracle e-Business component

1   Mount the file system.

2   Enable the IP address assigned to the component.

3   Run the Oracle supplied installer:

```
$ rapidwiz -servername <virtual hostname>
```

where:

<virtual hostname> is the virtual hostname of the components instance.

4   Follow the instructions mentioned on the wizard.

Configuring Oracle e-Business components for cluster support

After installing the Oracle e-Business components, the VCS agent for Oracle e-Business components manages and monitors individual components. In case of Oracle Process Manager and Notification Server (OPMN) it is thus required to disable its default behavior; automatic restart of its managed processes (OAFM, OACORE, HTTP_Server and FormsServer), during its failure. To disable this behavior you must configure the OPMN daemon by disabling the restart-on-death option for components managed by VCS within the opmn.xml Oracle Application Server configuration file.

Note: To view the list of OPMN managed components refer to the Oracle Application Server configuration file `opmn.xml`. This file is located at $INST_TOP/ora/10.1.3/opmn/conf/

For components installed, modify the `opmn.xml` file and from the process-set tag of each ias-component set the restart-on-death parameter to 'False'.

For example, to modify this setting for a HTTP_server and Forms component, refer to the following configuration details.

Note: The example shows a sample section of the `opmn.xml` file. It illustrates the changes that must be made for every OPMN managed component that is managed under VCS.

```
<ias-component id="HTTP_Server">
            <process-type id="HTTP_Server" module-id="OHS">
              <environment>
                    <variable id="DEFAULT_LOGS_DIR" value= \
                     "/ebiz/oracle/inst/apps/ \
                     VIS_ebiz2/logs/ora/10.1.3/Apache" \
                     append="true"/>
              </environment>

            <module-data>
              <category id="start-parameters">
                 <data id="start-mode" value="ssl-enabled"/>
                 <data id="config-file" value="/ebiz/oracle/ \
                  inst/apps/VIS_ebiz2/ora/ \
                  10.1.3/Apache/Apache/conf/httpd.conf"/>
```

```
                              </category>
                        </module-data>
                        <process-set id="HTTP_Server" restart-on-death="false" \
                         numprocs="1"/>
                    </process-type>
              </ias-component>
              <ias-component id="OC4J">
                  <process-type id="forms" module-id="OC4J" status="enabled" \
                   working-dir="$ORACLE_HOME/j2ee/home">
                      <module-data>
                          <category id="start-parameters">
                              <data id="java-options" value="-server -verbose:gc \
                               -Xmx256M -Xms64M -XX:MaxPermSize=128M \
                               -XX:NewRatio=2 \
                               -XX:+PrintGCTimeStamps -XX:+UseT
LAB -XX:+UseParallelGC -XX:ParallelGCThreads=2 -Djava.security.policy= \
$ORACLE_HOME/j2ee/oacore/config/java2.policy -Djava.awt.headless=true \
-Dhttp.webdir.ena
ble=false -Doracle.security.jazn.config=/ebiz/oracle/ \
inst/apps/VIS_ebiz2/ora/10.1.3/j2ee/forms/config/jazn.xml"/>
                              <data id="java-bin" value="/ebiz/oracle/inst/apps/ \
                               VIS_ebiz2/admin/scripts/java.sh"/>
                              <data id="oc4j-options" value="-out /ebiz/oracle/ \
                               inst/apps/VIS_ebiz2/logs/ora/10.1.3/opmn/ \
                               formsstd.out -err /ebiz/oracle/inst/apps/VIS_e
biz2/logs/ora/10.1.3/opmn/formsstd.err" />
                          </category>
                          <category id="stop-parameters">
                              <data id="java-options" value="-server -verbose:gc \
                               -Xmx256M -Xms64M -XX:MaxPermSize=128M \
                               -XX:NewRatio=2 \
                               -XX:+PrintGCTimeStamps -XX:+UseT
LAB -XX:+UseParallelGC -XX:ParallelGCThreads=2 -Djava.security.policy= \
$ORACLE_HOME/j2ee/oacore/config/java2.policy -Djava.awt.headless=true \
-Dhttp.webdir.ena
ble=false"/>
                          </category>
                        </module-data>
                        <start timeout="600" retry="2"/>
                        <stop timeout="120"/>
                        <restart timeout="720" retry="2"/>
                        <port id="default-web-site" range="22000-22004" \
                         protocol="ajp"/>
```

```
            <port id="rmi" range="20500-20504"/>
            <port id="jms" range="23500-23504"/>
            <process-set id="default_group" restart-on-death= \
             "false" numprocs="1"/>
        </process-type>
   </ias-component>
```

See "Installing Oracle e-Business components for clustering purposes" on page 33.

Troubleshooting the agent for Oracle e-Business Components

This chapter includes the following topics:

- Using correct software and operating system versions
- Meeting prerequisites
- Configuring Oracle e-Business Components resources
- Starting the Oracle e-Business Components instance outside a cluster
- Reviewing error log files

Using correct software and operating system versions

Ensure that no issues arise due to incorrect software and operating system versions. For the correct versions of operating system and software to be installed on the resource systems:

See "Supported software" on page 10.

Meeting prerequisites

Before installing the agent for Oracle e-Business Components, double check that you meet the prerequisites.

For example, you must install the ACC library on VCS before installing the agent for Oracle e-Business Components.

See "Before you install the Veritas agent for Oracle e-Business Components" on page 17.

Configuring Oracle e-Business Components resources

Before using an e-Business component resource, ensure that you configure the resource properly.

Refer to the agent attributes for the list of resource types with which you can configure the component resource.

For information about setting the agent attributes review the following:

■ About sample configurations for the agent for Oracle e-Business Components

Starting the Oracle e-Business Components instance outside a cluster

If you face problems while working with a resource, you must disable the resource within the cluster framework. A disabled resource is not under the control of the cluster framework, and so you can test the Oracle e-Business Components instance independent of the cluster framework. Refer to the cluster documentation for information about disabling a resource.

You can then restart the Oracle e-Business Components instance outside the cluster framework.

Note: Use the same parameters that the resource attributes define within the cluster framework while restarting the resource outside the cluster framework.

A sample procedure to start a appshort instance outside the cluster framework, is illustrated as follows.

To restart the resource outside the framework

1   Ensure that you freeze the service group, so that the cluster does not take control of the resource running the e-Business component instance.

2   Start the component using the following Oracle supplied scripts:

Oracle component for Oracle Apps 11i	Oracle script to start or stop the component	List of processes
Forms Server	adfrmctl.sh	f60srvm, f60webmx

Forms Metric Server	adfmsctl.sh	d2ls60
Forms Metric Client	adfmcctl.sh	d2lc60
Reports Server	adrepctl.sh	rwmts60
Web Server	adapcctl.sh	httpd
Discoverer	addisctl.sh	osagent, oad
RPC Listener	adalnctl.sh	tnslsnr
Fulfillment Server	jtffmctl.sh	java.*FulfillmentServer*

Oracle component for Oracle Apps R12	Oracle script to start or stop the component	List of processes
OPMN	adopmnctl.sh	opmn
Forms Server (Servlet Mode)	adformsctl.sh	forms
Forms Server (Socket Mode)	adformsrvctl.sh	frmsrv
OACORE Server	adcorectl.sh	oacore
OAFM Server	adoafmctl.sh	oafm
Web Server	adapcctl.sh	httpd
RPC Listener	adalnctl.sh	tnslsnr
Fulfillment Server	jtffmctl.sh	java.*FulfillmentServer*

For example, to start a Forms Server component instance, run this command:

For Oracle Apps 11i

```
# adfrmctl.sh start
```

For Oracle Apps R12

```
# adformsctl.sh start
```

These scripts are located in the $COMMON_TOP/admin/scripts/*CONTEXT_NAME*/ directory for the Oracle Apps 11i and in the $INST_TOP/admin/scripts directory for the Oracle Apps R12.

Where, *CONTEXT_NAME* is the name of the context file. The format of the context file is SID_*hostname*

Ensure that the component instance starts successfully. If the instance works properly outside the cluster framework, you can attempt to implement the instance within the framework.

To stop the resource outside the framework

1 Ensure that you freeze the service group, so that the cluster does not take control of the resource running the e-Business component instance.

2 Stop the component using the Oracle supplied scripts.

For example, to stop a Forms Server component instance, run this command:

For Oracle Apps 11i

```
# adfrmctl.sh stop
```

For Oracle Apps R12

```
# adformsctl.sh stop
```

These scripts are located in the $COMMON_TOP/admin/scripts/*CONTEXT_NAME*/ directory for the Oracle Apps 11i and in the $INST_TOP/admin/scripts directory for the Oracle Apps R12.

Where, *CONTEXT_NAME* is the name of the context file. The format of the context file is SID_*hostname*

Ensure that the component instance stops successfully. If the instance works properly outside the cluster framework, you can attempt to implement the instance within the framework.

Reviewing error log files

If you face problems while using Oracle e-Business Components or the agent for Oracle e-Business Components, use the log files described in this section to investigate the problems.

Reviewing the screen output

While starting or stopping the Oracle e-Business CM instance, you can review the command output displayed on the screen to analyze the problem.

Reviewing cluster log files

In case of problems while using the agent for Oracle e-Business Components, you can access the engine log file for more information about a particular resource. The engine log file is located at /var/VRTSvcs/log/engine_A.log.

Reviewing agent log files

You can access the $COMMON_TOP/admin/log directory for Oracle Apps 11i and $INST_TOP/logs directory for Oracle Apps R12 and view the log files to understand the problem while using the agent for Oracle e-Business components.

Using trace level logging

The ResLogLevel attribute controls the level of logging that is written in a cluster log file for each Oracle e-Business Components resource. You can set this attribute to TRACE, which enables very detailed and verbose logging.

If you set ResLogLevel to TRACE, a very high volume of messages are produced. Symantec recommends that you localize the ResLogLevel attribute for a particular resource.

Note: Starting with version 5.1.1.0 of the ACC library, the TRACE level logs for any ACCLib based agent are generated locally at the location /var/VRTSvcs/log/*Agent*_A.log.

To localize ResLogLevel attribute for a resource

1   Identify the resource for which you want to enable detailed logging.

2   Localize the ResLogLevel attribute for the identified resource:

    # hares -local *Resource_Name* ResLogLevel

3   Set the ResLogLevel attribute to TRACE for the identified resource:

    # hares -modify *Resource_Name* ResLogLevel TRACE -sys *SysA*

4   Note the time before you begin to operate the identified resource.

5   Test the identified resource. The function reproduces the problem that you are attempting to diagnose.

6   Note the time when the problem is reproduced.

7 Set the ResLogLevel attribute back to INFO for the identified resource:

```
# hares -modify Resource_Name ResLogLevel INFO -sys SysA
```

8 Review the contents of the log file. Use the time noted in Step 4 and Step 6 to diagnose the problem.

You can also contact Symantec support for more help.

Sample Configurations

This appendix includes the following topics:

-

-

-

-

About sample configurations for the agent for Oracle e-Business Components

The sample configuration graphically depicts the resource types, resources, and resource dependencies within the service group. Review these dependencies carefully before configuring the agent for Oracle e-Business Components. For more information about these resource types, see the *Veritas Cluster Server Bundled Agents Reference Guide*.

Sample agent type definition

The sample agent type definition for Oracle e-Business Components is as follows:

For VCS 4.x

```
type OracleApps (
 static str ArgList[] = { ResLogLevel, State, IState,
                          User, OracleAppsVersion, OracleHome,
                          ScriptHome,
                          EnvFile,ServerType, Port,
                          DBConnectString,
```

```
                                SecondLevelMonitor,
                                MonitorProgram }
 str ResLogLevel = INFO
 str User
 int OracleAppsVersion
 str OracleHome
 str ScriptHome
 str EnvFile
 str ServerType
 str Port
 str DBConnectString
 int SecondLevelMonitor = 0
 str MonitorProgram
)
```

For VCS 5.0

```
type OracleApps (
        static str AgentFile = "/opt/VRTSvcs/bin/Script50Agent"
        static str AgentDirectory = "/opt/VRTSagents/ha/bin/OracleApps"
        static str ArgList[] = { ResLogLevel, State, IState, User,
        OracleAppsVersion, OracleHome, ScriptHome, EnvFile,
        ServerType, Port, DBConnectString, SecondLevelMonitor,
        MonitorProgram, VirtualHostname}
        str ResLogLevel = INFO
        str User
        int OracleAppsVersion
        str OracleHome
        str ScriptHome
        str EnvFile
        str ServerType
        str Port
        str DBConnectString
        int SecondLevelMonitor = 0
        str MonitorProgram
        str VirtualHostname
)
```

Sample Oracle HTTP Server instance

An excerpt of the main.cf file for an Oracle HTTP Server instance is as follows.

For Oracle Apps 11i

```
OracleApps OraApps_HTTPSrvr
(
OracleAppsVersion    = 11
User                 = oraweb
ScriptHome           = "/ora_apps/web/viscomn/admin/scripts/VIS"
ServerType           = WebServer
Port                 = 8002
EnvFile              = "VIS.env"
SecondLevelMonitor   = 5
)
```

For Oracle Apps R12

```
OracleApps OraApps_HTTPSrvr
(
OracleAppsVersion    = 12
User                 = applmgr
ORACLE_HOME          = "/ebiz/oracle/VIS/apps/tech_st/10.1.3"
ScriptHome           = "/ebiz/oracle/inst/apps/VIS_ebiz1/admin/scripts"
ServerType           = WebServer
EnvFile              = "/ebiz/oracle/inst/apps/VIS_ebiz1/ora/10.1.3/\
                       VIS_ebiz.env"
SecondLevelMonitor   = 5
)
```

Sample Oracle OPMN instance

```
OracleApps OraApps_OPMN
(
OracleAppsVersion    = 12
ORACLE_HOME          = "/ebiz/oracle/VIS/apps/tech_st/10.1.3"
ScriptHome           = "/ebiz/oracle/inst/apps/VIS_ebiz1/admin/scripts"
ServerType           = OPMN
User                 = applmgr
SecondLevelMonitor   = 5
EnvFile              = "/ebiz/oracle/inst/apps/VIS_ebiz1/ora/10.1.3/\
                       VIS_ebiz.env"
)
```

Sample Oracle OACORE instance

```
OracleApps OraApps_OACORE
(
```

```
OracleAppsVersion    = 12
ORACLE_HOME          = "/ebiz/oracle/VIS/apps/tech_st/10.1.3"
ScriptHome           = "/ebiz/oracle/inst/apps/VIS_ebiz1/admin/scripts"
ServerType           = OACORE
User                 = applmgr
SecondLevelMonitor   = 5
EnvFile              = "/ebiz/oracle/inst/apps/VIS_ebiz1/ora/10.1.3/\
                        VIS_ebiz.env"
)
```

Sample Oracle OAFM instance

```
OracleApps OraApps_OAFM
(
OracleAppsVersion    = 12
ORACLE_HOME          = "/ebiz/oracle/VIS/apps/tech_st/10.1.3"
ScriptHome           = "/ebiz/oracle/inst/apps/VIS_ebiz1/admin/scripts"
ServerType           = OAFM
User                 = applmgr
SecondLevelMonitor   = 5
EnvFile              = "/ebiz/oracle/inst/apps/VIS_ebiz1/ora/10.1.3/\
                        VIS_ebiz.env"
)
```

Sample Oracle Forms Listener instance

An excerpt of the main.cf file for an Oracle Forms Listener instance is as follows.

```
OracleApps oraforms_lsnr
(
OracleAppsVersion = 11
ScriptHome        = "/ora_apps/forms/viscomn/admin/scripts/VIS"
ServerType        = Listener
User              = oraforms
ORACLE_HOME       = "/ora_apps/forms/visora/8.0.6"
EnvFile           = "VIS.env"
)
```

Sample Oracle Forms Server instance

An excerpt of the main.cf file for an Oracle Forms Server instance is as follows.

For Oracle Apps 11i

```
OracleApps OraApps_FormsSrvr
(
OracleAppsVersion  = 11
ScriptHome         = "/ora_apps/forms/viscomn/admin/scripts/VIS"
ServerType         = FormsServer
User               = oraforms
SecondLevelMonitor = 5
)
```

For Oracle Apps R12

```
OracleApps OraApps_FormsSrvr
(
OracleAppsVersion  = 12
ORACLE_HOME        = "/ebiz/oracle/VIS/apps/tech_st/10.1.3"
ScriptHome         = "/ebiz/oracle/inst/apps/VIS_ebiz1/admin/scripts"
ServerType         = FormsServer
User               = applmgr
SecondLevelMonitor = 5
EnvFile            = "/ebiz/oracle/inst/apps/VIS_ebiz1/ora/10.1.3/\
                      VIS_ebiz.env"
)
```

For Oracle Apps 11i, the dependency of resources is as follows:

- OraApps_HTTPSrvr requires OraApps_ip

- OraApps_HTTPSrvr requires OraApps_mnt

- OraApps_FormsLsnr requires OraApps_FormsSrvr

- OraApps_FormsSrvr requires OraApps_ip

- OraApps_FormsSrvr requires OraApps_mnt

- OraApps_mnt requires OraApps_dg

For Oracle Apps R12, the dependency of resources is as follows:

- OraApps_FormsSrvr requires OraApps_OPMN

- OraApps_OACORE requires OraApps_OPMN

- OraApps_OAFM requires OraApps_OPMN

- OraApps_HTTPSrvr requires OraApps_OPMN

- OraApps_OPMN requires OraApps_ip

- OraApps_OPMN requires OraApps_mnt

- OraApps_mnt requires OraApps_dg

Sample agent type definitions with Solaris zone support

The sample agent type definition for Oracle e-Business Components with Solaris zone support is as follows:

For VCS 4.x

```
type OracleApps (
        static str ContainerType = Zone
        static str ArgList[] = { ResLogLevel, State, IState, User,
        OracleAppsVersion, OracleHome, ScriptHome, EnvFile,
        ServerType, Port, DBConnectString, SecondLevelMonitor,
        MonitorProgram, VirtualHostname}
        str ResLogLevel = INFO
        str User
        int OracleAppsVersion
        str OracleHome
        str ScriptHome
        str EnvFile
        str ServerType
        str Port
        str DBConnectString
        int SecondLevelMonitor = 0
        str MonitorProgram
        str VirtualHostname
        str ContainerName
)
```

For VCS 5.0

```
type OracleApps (
        static str ContainerType = Zone
        static str AgentFile = "/opt/VRTSvcs/bin/Script50Agent"
        static str AgentDirectory = "/opt/VRTSagents/ha/bin/OracleApps"
        static str ArgList[] = { ResLogLevel, State, IState, User,
        OracleAppsVersion, OracleHome, ScriptHome, EnvFile,
        ServerType, Port, DBConnectString, SecondLevelMonitor,
        MonitorProgram, VirtualHostname}
        str ResLogLevel = INFO
        str User
        int OracleAppsVersion
        str OracleHome
        str ScriptHome
```

```
        str EnvFile
        str ServerType
        str Port
        str DBConnectString
        int SecondLevelMonitor = 0
        str MonitorProgram
        str VirtualHostname
        str ContainerName
)
```

Sample service group configuration

A service group is composed of different resources, that are related to each other and are formed along traditional Oracle e-Business functional lines. You can independently start, stop, and monitor each resource within a service group.

For example, you can group an instance each of a Forms Server, a Forms Listener, a Forms Metric Client, and a Forms Metric Server within a service group.

Symantec recommends that a service group must be configured using a virtual IP address and shared disk storage. All resources in a service group share the same virtual IP address, disk storage, and UNIX User account. In event of a failure, an application resource can run on any node in the cluster without being tied to a node specific IP address or to local disk resources.

The following figures presents sample configuration service groups for different components of Oracle e-Business 11i.

Note: For each sample service group, the Oracle e-Business components are dependent on underlying virtual IP address and shared disk resources to support local fail-over.

Figure A-1 shows a service group with Oracle 9iAS Web Application Server resource.

The resource depends on an IP and Mount resource. The application administrator must also configure an application "listener" for the web application server. In this case, the listener resource is dependent on the Web Application Server instance.

Figure A-1   Service group with Oracle 9iAS Web Application Server resource
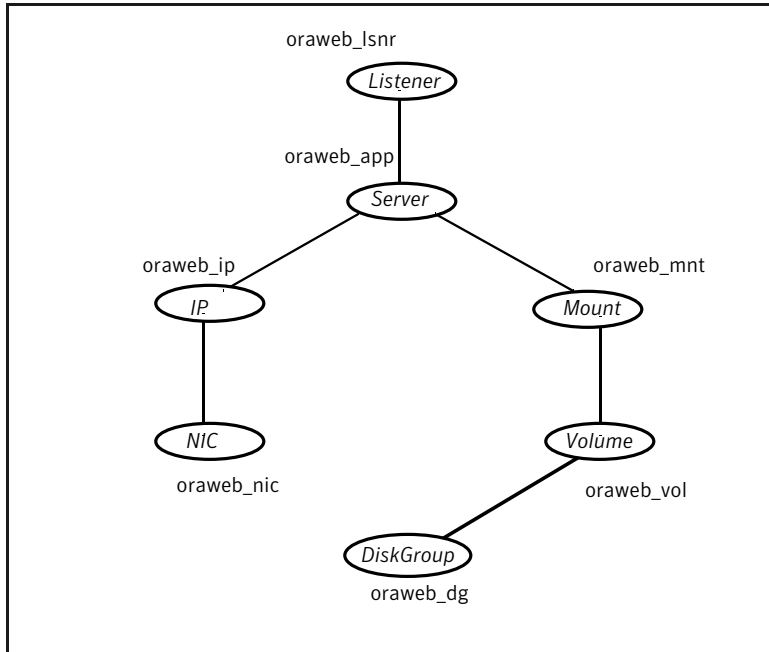
Figure A-2 depicts a service group that contains an Oracle Forms Server
component, that includes an instance each of a Forms Server, a Forms Metric
Client, a Forms Metric Server, and a Forms Listener.

A separate resource within the group controls each Oracle Forms Server
component. In this example, the Listener resource is dependent on the Forms
Server, and can easily be dependent directly on the underlying IP and mount
resources.

Figure A-2   Service group containing an Oracle Forms Server component
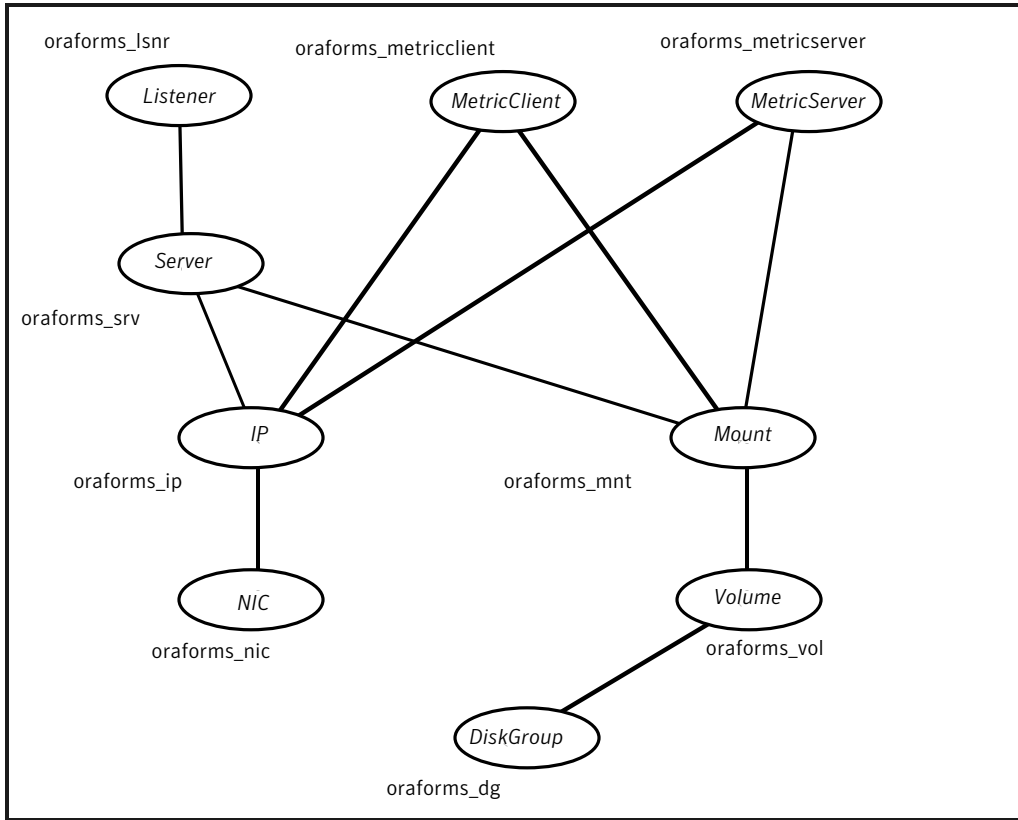
Figure A-3 depicts a common setup for the Oracle Report Server as a companion to the Oracle Concurrent Manager.

The Report Server can also be separately started, stopped, and monitored in this configuration.

Figure A-3   Common setup for the Oracle Report Server as a companion to the Oracle Concurrent Manager
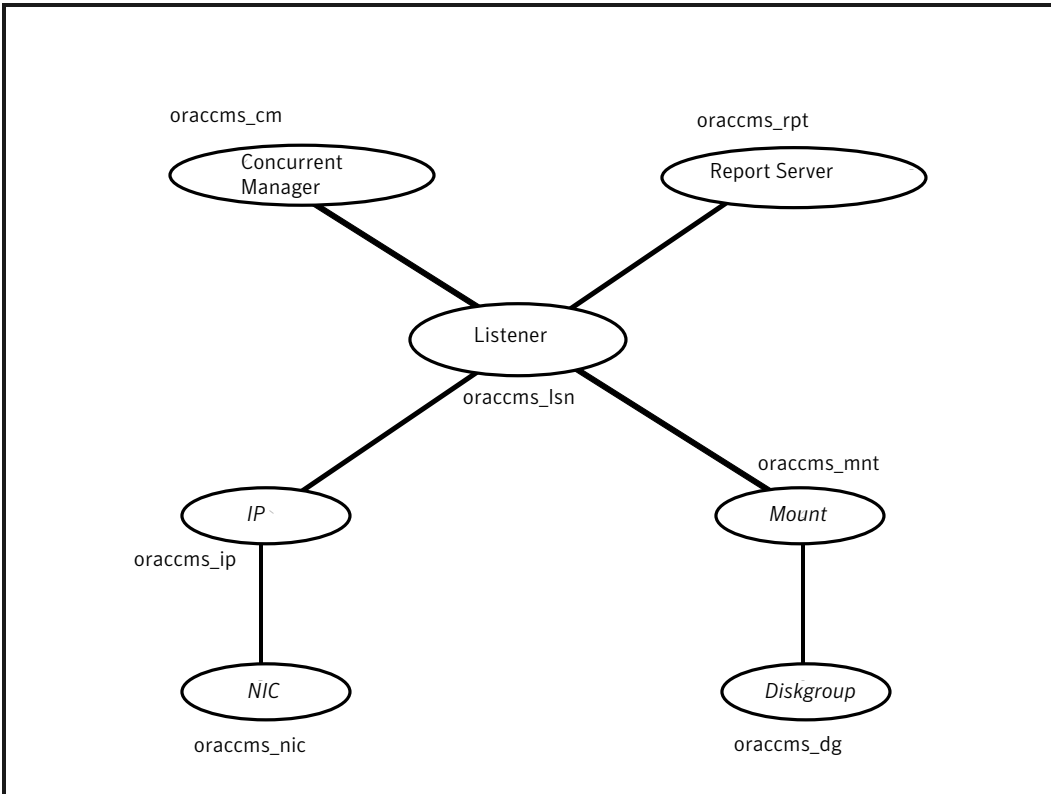
Figure A-4 depicts a setup for the Oracle Administration server service group.

The Administration server is installed as a separate component using the Rapidwiz installer. This configuration shows the use of a unique disk group and mount point dedicated to a single Oracle Administration server instance. The Administration server resource is configured as a "listener".

Figure A-4   Setup for the Oracle Administration Server service group
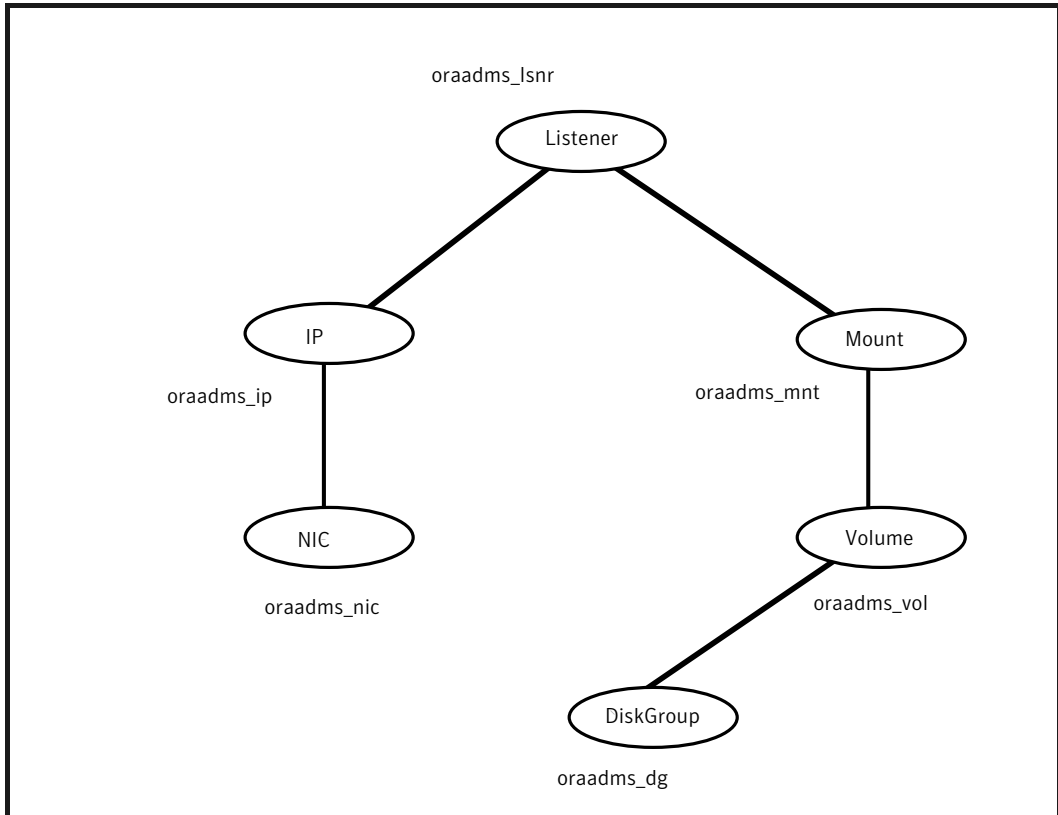
For Oracle e-Business R12, the OC4J forms replace the forms and reports server with the introduction of Application Server 10g. These forms are managed by Oracle Process Manager and Notification Server (OPMN)

The following figures present sample configuration service groups for different components of Oracle e-Business R12.

Figure A-5 depicts a service group with Oracle oAS 10g OPMN and its managed processes; Web Server, Forms, OAFM and OACORE.

Figure A-5   Service group with Oracle oAS 10g OPMN and its managed processes
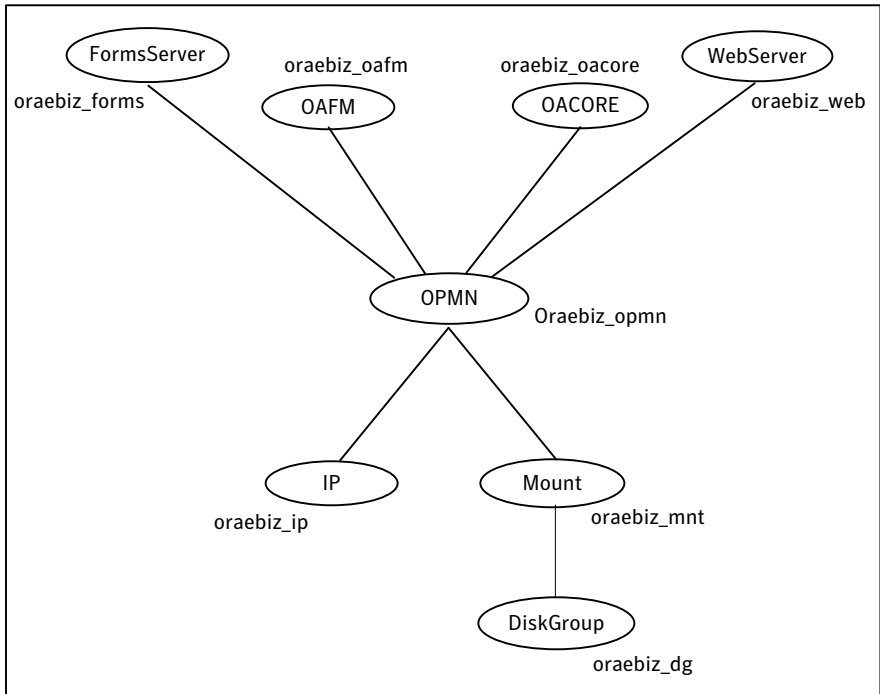
Figure A-6 depicts a service group with Oracle oAS 10g OPMN and its managed processes; Web Server, Forms, OAFM and OACORE, under Solaris zone.

Figure A-6 Service group with Oracle oAS 10g OPMN and its managed
processes, under Solaris zone

Index