

Cluster Server Agent for Oracle e-Business Components Installation and Configuration Guide

AIX, Linux, Solaris

7.0

Veritas InfoScale™ Availability Agents

Last updated: 2018-02-27

Legal Notice

Copyright © 2018 Veritas Technologies LLC. All rights reserved.

Veritas and the Veritas Logo are trademarks or registered trademarks of Veritas Technologies LLC or its affiliates in the U.S. and other countries. Other names may be trademarks of their respective owners.

This product may contain third-party software for which Veritas is required to provide attribution to the third party ("Third Party Programs"). Some of the Third Party Programs are available under open source or free software licenses. The License Agreement accompanying the Software does not alter any rights or obligations you may have under those open source or free software licenses. Refer to the third-party legal notices document accompanying this Veritas product or available at:

<https://www.veritas.com/about/legal/license-agreements>

The product described in this document is distributed under licenses restricting its use, copying, distribution, and decompilation/reverse engineering. No part of this document may be reproduced in any form by any means without prior written authorization of Veritas Technologies LLC and its licensors, if any.

THE DOCUMENTATION IS PROVIDED "AS IS" AND ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT SUCH DISCLAIMERS ARE HELD TO BE LEGALLY INVALID. VERITAS TECHNOLOGIES LLC SHALL NOT BE LIABLE FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES IN CONNECTION WITH THE FURNISHING, PERFORMANCE, OR USE OF THIS DOCUMENTATION. THE INFORMATION CONTAINED IN THIS DOCUMENTATION IS SUBJECT TO CHANGE WITHOUT NOTICE.

The Licensed Software and Documentation are deemed to be commercial computer software as defined in FAR 12.212 and subject to restricted rights as defined in FAR Section 52.227-19 "Commercial Computer Software - Restricted Rights" and DFARS 227.7202, et seq. "Commercial Computer Software and Commercial Computer Software Documentation," as applicable, and any successor regulations, whether delivered by Veritas as on premises or hosted services. Any use, modification, reproduction release, performance, display or disclosure of the Licensed Software and Documentation by the U.S. Government shall be solely in accordance with the terms of this Agreement.

Veritas Technologies LLC
500 E Middlefield Road
Mountain View, CA 94043

<http://www.veritas.com>

Technical Support

Technical Support maintains support centers globally. All support services will be delivered in accordance with your support agreement and the then-current enterprise technical support policies. For information about our support offerings and how to contact Technical Support, visit our website:

<https://www.veritas.com/support>

You can manage your Veritas account information at the following URL:

<https://my.veritas.com>

If you have questions regarding an existing support agreement, please email the support agreement administration team for your region as follows:

Worldwide (except Japan)

CustomerCare@veritas.com

Japan

CustomerCare_Japan@veritas.com

Documentation

Make sure that you have the current version of the documentation. Each document displays the date of the last update on page 2. The latest documentation is available on the Veritas website:

<https://sort.veritas.com/documents>

Documentation feedback

Your feedback is important to us. Suggest improvements or report errors or omissions to the documentation. Include the document title, document version, chapter title, and section title of the text on which you are reporting. Send feedback to:

xyz@veritas.com

You can also see documentation information or ask a question on the Veritas community site:

<http://www.veritas.com/community/>

Veritas Services and Operations Readiness Tools (SORT)

Veritas Services and Operations Readiness Tools (SORT) is a website that provides information and tools to automate and simplify certain time-consuming administrative tasks. Depending on the product, SORT helps you prepare for installations and upgrades, identify risks in your datacenters, and improve operational efficiency. To see what services and tools SORT provides for your product, see the data sheet:

https://sort.veritas.com/data/support/SORT_Data_Sheet.pdf

Contents

| | | |
|------------------|--|-----------|
| Chapter 1 | Introducing the agent for Oracle e-Business Components | 7 |
| | About the Cluster Server agent for Oracle e-Business Components | 7 |
| | Supported software | 9 |
| | How the agent makes Oracle e-Business Components highly available | 9 |
| | High availability for Oracle e-Business Components instances | |
| | running in Solaris zones | 9 |
| | Oracle e-Business Components agent functions | 10 |
| | Online | 10 |
| | Offline | 10 |
| | Monitor | 11 |
| | Clean | 11 |
| | Detecting application failure | 11 |
| | Setting up Oracle e-Business Components in a VCS cluster | 15 |
| | | |
| Chapter 2 | Installing and configuring Oracle e-Business Components for high availability | 16 |
| | About Oracle e-Business Components | 16 |
| | About configuring Oracle e-Business Components for high availability | 17 |
| | Installing Oracle e-Business components for clustering purposes | 17 |
| | Decide the type of installation as single-node or multiple-node | 17 |
| | Configure a disk group and a file system | 18 |
| | Obtain virtual IP addresses for each node | 18 |
| | Install the Oracle e-Business component | 18 |
| | Migrating existing installation of Oracle e-Business Components from | |
| | physical to virtual hostname | 19 |
| | Configuring Oracle e-Business components for cluster support | 19 |
| | Configuring the OracleApps resource for Fulfillment Server | 22 |
| | Setting up zones on Solaris for Oracle e-Business Components | 23 |

| | | |
|------------------|---|-----------|
| Chapter 3 | Installing, upgrading, and removing the agent for Oracle e-Business Components | 26 |
| | Before you install the Cluster Server agent for Oracle e-Business Components | 26 |
| | Prerequisites for installing the agent to support Solaris zones | 27 |
| | About the ACC library | 27 |
| | Installing the ACC library | 28 |
| | Installing the ACC library IPS package on Oracle Solaris 11 systems | 28 |
| | Installing the ACC library package on Solaris brand non-global zones | 29 |
| | Installing the agent in a VCS environment | 30 |
| | Installing the agent IPS package on Oracle Solaris 11 systems | 32 |
| | Installing agent packages on Solaris brand non-global zones | 32 |
| | Installing the agent in a Solaris 10 brand zone | 33 |
| | Uninstalling the agent in a VCS environment | 33 |
| | Removing the ACC library | 35 |
| | Upgrading the agent for OracleApps | 35 |
| Chapter 4 | Configuring the agent for Oracle e-Business Components | 38 |
| | About configuring the Cluster Server agent for Oracle e-Business Components | 38 |
| | Importing the agent types files in a VCS environment | 39 |
| | Agent attributes for Oracle e-Business components | 40 |
| | Executing a customized monitoring program | 48 |
| | Creating soft links to JDBC properties file | 48 |
| Chapter 5 | Configuring the service groups for Oracle e-Business Components using the CLI | 50 |
| | About configuring service groups for Oracle e-Business Components | 50 |
| | Before configuring the service groups for Oracle e-Business Components | 51 |
| | Configuring Oracle e-Business Components resources for Solaris zones support | 51 |

| | | |
|--------------------|--|-----------|
| Chapter 6 | Troubleshooting the agent for Oracle e-Business Components | 52 |
| | Using the correct software and operating system versions | 52 |
| | Meeting prerequisites | 52 |
| | Configuring Oracle e-Business Components resources | 53 |
| | Starting the Oracle e-Business Components instance outside a cluster | 53 |
| | Reviewing error log files | 57 |
| | Reviewing the screen output | 57 |
| | Reviewing cluster log files | 58 |
| | Reviewing agent log files | 58 |
| | Using trace level logging | 58 |
| Appendix A | Sample Configurations | 60 |
| | About sample configurations for the agents for Oracle e-Business Components | 60 |
| | Sample configuration file for Oracle e-Business Concurrent Manager for Linux | 60 |
| | Sample agent type definition | 62 |
| | Sample Oracle OPMN instance | 63 |
| | Sample Oracle HTTP Server instance | 63 |
| | Sample Oracle OACORE instance | 64 |
| | Sample Oracle OAFM instance | 64 |
| | Sample Oracle Forms Listener instance | 65 |
| | Sample Oracle Forms Server instance | 65 |
| | Sample agent type definitions with Solaris zone support | 67 |
| | Sample Oracle Application Server resource configuration | 68 |
| | Sample service group configuration | 76 |
| Index | | 84 |

Introducing the agent for Oracle e-Business Components

This chapter includes the following topics:

- [About the Cluster Server agent for Oracle e-Business Components](#)
- [Supported software](#)
- [How the agent makes Oracle e-Business Components highly available](#)
- [Oracle e-Business Components agent functions](#)
- [Detecting application failure](#)
- [Setting up Oracle e-Business Components in a VCS cluster](#)

About the Cluster Server agent for Oracle e-Business Components

Cluster Server (VCS) agents monitor specific resources within an enterprise application. They determine the status of resources and start or stop them according to external events.

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

The Cluster Server 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 11i, R12, and R12.2 releases of Oracle e-Business Component.

The agent manages the following Oracle e-Business components:

| Version | Components |
|---------|--|
| R12.2 | <ul style="list-style-type: none"> ■ Administration Listener ■ Fulfillment Server ■ HTTP Server ■ OPMN ■ WebLogic AdminServer ■ WebLogic NodeManager ■ WebLogic Forms Server ■ WebLogic OACORE Managed Server ■ WebLogic OAFM Managed Server |
| R12 | <ul style="list-style-type: none"> ■ Administration Listener ■ OPMN ■ OC4J Forms Server ■ OC4J OACORE ■ OC4J OAFM ■ HTTP Server ■ Fulfillment Server |
| 11i | <ul style="list-style-type: none"> ■ 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 the Discoverer 10g component, you must install the Oracle AS9 agent.

For the latest updates or software issues for this agent, see the *Cluster Server Agent Pack Release Notes*.

Supported software

For information on the software versions that the Cluster Server agent for Oracle e-Business Components supports, see the Veritas Services and Operations Readiness Tools (SORT) site: <https://sort.veritas.com/agents>.

How the agent makes Oracle e-Business Components highly available

The agent provides the following levels of application monitoring:

- Primary or Basic monitoring
This mode has Process check and Health check monitoring options. With the default Process check option, the agent verifies that the Oracle e-Business Components instance processes are present in the process table. Process check cannot detect whether processes are in the hung or stopped states.
- Secondary or Detail monitoring
In this mode, the agent runs a utility to verify the status of the Oracle e-Business Components instance. The agent detects application failure if the monitoring routine reports an improper function of the Oracle e-Business Components instance processes. When this application failure occurs, the Oracle e-Business Components instance service group fails over to another node in the cluster. Thus, the agent ensures high availability for Oracle e-Business Components instances.

High availability for Oracle e-Business Components instances running in Solaris zones

Solaris provides a means of virtualizing operating system services, allowing one or more processes to run in isolation from other activity on the system. Such a 'sandbox' is called a 'non-global zone'. Each zone can provide a rich and customized set of services. The processes that run in a 'global zone' have the same set of privileges that are available on a Solaris system today.

VCS provides high availability to applications running in non-global zones by extending the failover capability to zones. VCS is installed in a global zone, and all the agents and the engine components run in the global zone. For applications running within non-global zones, agents run script entry points inside the zones. If a zone configured under VCS control faults, VCS fails over the entire service group containing the zone.

For more details, refer to the *Cluster Server Administrator's Guide*.

The Cluster Server agent for Oracle e-Business Components is zone-aware and can monitor Oracle e-Business Components instances running in non-global zones.

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 11.

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 11.

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 the first and second level monitor functions work.
See [“Detecting application failure”](#) on page 11.
- 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 48.

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.

The following table 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 | adfmctl.sh | d2ls60 |
| Forms Metric Client | adfmctl.sh | d2lc60 |
| Reports Server | adrepctl.sh | rwmts60 |
| Web Server | adapctl.sh | httpd |
| Discoverer | addisctl.sh | osagent, oad |
| RPC Listener | adalnctl.sh | tnslsnr |
| Fulfillment Server | jtffmctl.sh | java.*FulfillmentServer* |

The following table 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 |
| OACORE Server | adcorectl.sh | oacore |
| OAFM Server | adoafmctl.sh | oafm |

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 |
|--------------------|--|--------------------------|
| Web Server | adapctl.sh | httpd |
| RPC Listener | adalctl.sh | tnslsnr |
| Fulfillment Server | jtffmctl.sh | java.*FulfillmentServer* |

The following table lists the scripts and process for Oracle e-Business R12.2 component instances

Table 1-3 Scripts and processes for Oracle e-Business R12.2 component instances

| Oracle component | Oracle script to start or stop the component | List of processes |
|-----------------------------|--|--------------------------|
| AdminServer | adadminsrvctl.sh | AdminServer |
| NodeManager | adnodemgrctl.sh | weblogic. NodeManager |
| OPMN | adopmnctl.sh | opmnsh |
| Forms Server (Servlet Mode) | admanagedsrvctl.sh | ServerName |
| Forms Server (Socket Mode) | adformsrvctl.sh | frmsrv |
| OACORE Server | admanagedsrvctl.sh | ServerName |
| OAFM Server | admanagedsrvctl.sh | ServerName |
| Web Server | adapctl.sh | httpd |
| RPC Listener | adalctl.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 `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 `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 the `opmnctl status` test, where `opmnctl` is a command provided by oAS 10g to verify the `opmn` and managed processes.
- For EBS 12.2:

If the `ServerType` value is ...the monitor function performs this test

| | |
|--------------------------|--------------------------------------|
| <code>AdminServer</code> | <code>adadminsrvctl.sh status</code> |
| <code>NodeManager</code> | <code>adnodemgrctl.sh status</code> |
| <code>WebServer</code> | <code>adapcctl.sh status</code> |
| <code>opmn</code> | <code>adopmnctl.sh status</code> |
| <code>Listener</code> | <code>adalnctl.sh status</code> |

- 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:

```
OC4J:oafm      | Alive
OC4J:forms     | Alive
OC4J:ocore     | Alive
HTTP_Server   | Alive
```

Note: OPMN does not manage Forms 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.

The agent supports multiple instances of Forms Server, OAFM Server, OACORE Server, and Fulfillment Server. The agent uses the Oracle Home directory to uniquely map multiple OracleApps resources with the appropriate server processes.

Setting up Oracle e-Business Components in a VCS cluster

Follow the steps below to set up Oracle e-Business Components in a cluster:

- Set up a VCS cluster.
For more information on installing and configuring Cluster Server, refer to the Cluster Server installation and configuration guides.
- Install and configure Oracle e-Business Components for High Availability.
See [“About configuring the Cluster Server agent for Oracle e-Business Components”](#) on page 38.
- Install the Cluster Server agent for Oracle e-Business Components.
See [“Installing the agent in a VCS environment”](#) on page 30.
- Configure the service groups for Oracle e-Business Components.
See [“About configuring service groups for Oracle e-Business Components”](#) on page 50.

Installing and configuring Oracle e-Business Components for high availability

This chapter includes the following topics:

- [About Oracle e-Business Components](#)
- [About configuring Oracle e-Business Components for high availability](#)
- [Installing Oracle e-Business components for clustering purposes](#)
- [Migrating existing installation of Oracle e-Business Components from physical to virtual hostname](#)
- [Configuring Oracle e-Business components for cluster support](#)
- [Configuring the OracleApps resource for Fulfillment Server](#)
- [Setting up zones on Solaris for Oracle e-Business Components](#)

About Oracle e-Business Components

An Oracle ERP application comprises various functional modules, each catering to an individual enterprise business function.

Oracle eBusiness Components serves as the Application Server Level components, which provides the base for these functional modules.

For example, HTTP_Server provides the Web Server for hosting the Oracle e-Business Web Interface, Form Server provides the Oracle Forms Interface, the Administration Listener provides the communication channel between Database and Apps Tier, and so on.

About configuring Oracle e-Business Components for high availability

The guidelines for configuring Oracle e-Business Components for high availability are as follows:

- In a service group, keep the single point of failure as minimal as possible and watch the application startup time.
- Assign a virtual hostname to the component within the switchover environment. Because the physical hostname changes with the switchover, this is a must-have requirement.
- Based on the expected failover time, configure the reconnection parameters for all software components and enable its automatic reconnection.

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.

Migrating existing installation of Oracle e-Business Components from physical to virtual hostname

The Oracle e-Business Suite is bound to the hostname, and not to the IP address. There are two ways to migrate the pre-installed and configured Oracle e-Business Suite instance from a physical to a virtual hostname

1. Clone the existing instance with the virtual hostname using the Oracle-provided 'Rapid Clone' utility. For more information about the procedure for Rapid Cloning of Oracle e-Business Suite Release 11i and Release 12, see the following Oracle metalink documents:
 - Cloning Oracle Applications Release 11i with Rapid Clone [ID 230672.1]
 - Cloning Oracle Applications Release 12 with Rapid Clone [ID 406982.1]
 - Adding an Alias Hostname with Oracle E-Business Suite Release 12 [Id 603883.1]
2. Rename the physical hostname, that is, use the existing hostname as the virtual hostname and change the physical hostname. In this case, you need to re-configure VCS to use the new physical hostname.

Note: Before bringing the application under VCS control, verify that application is running as before.

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 describes 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 17.

Configuring the OracleApps resource for Fulfillment Server

If you plan to configure the OracleApps resource for Fulfillment Server, you must modify the `jtffmctl.sh` script in the `ScriptHome` directory, before you start the Fulfillment Server process. This ensures that the agent uses the `OracleHome` attribute to uniquely identify the correct Fulfillment Server among multiple instances.

An extract from the `main.cf` file is as follows:

```
OracleApps fulfillment_server (
    Critical = 0
    ResLogLevel = TRACE
    User = applmgr
    OracleAppsVersion = 12
    OracleHome = "/oraem/oraebiz/oracle/VIS/apps/tech_st/10.1.2"
    ScriptHome = "/oraem/oraebiz/oracle/VIS/inst/apps/VIS_oraebiz/
        admin/scripts"
    EnvFile = "/oraem/oraebiz/oracle/VIS/apps/apps_st/appl/
        APPSVIS_oraebiz.env"
    ServerType = FulfillmentServer
    VirtualHostname = oraebiz
)
```

To configure the OracleApps resource for Fulfillment Server, you must modify the `jtffmctl.sh` script and add the following text before the `-Dengine.ServerID` string:

```
<space>-Doracle.home=/oraem/oraebiz/oracle/VIS/apps/tech_st
/10.1.2<space>
```

For example:

The `jtffmctl.sh` script is available

at: `/oraem/oraebiz/oracle/VIS/inst/apps/VIS_oraebiz/admin/scripts/jtffmctl.sh`.

An extract from the modified script is as follows:

```
_oraebiz/appl/fnd/12.0.0/secure/VIS.dbc -Doracle.home=/oraem
/oraebiz/oracle/VIS/apps/tech_st/10.1.2 -Dengine.ServerID=5000 -Ddebug=off...
```

Setting up zones on Solaris for Oracle e-Business Components

An example of creating a zone for Oracle e-Business Components on Solaris is as follows:

Step 1: Create the zone.

```
bash-3.00# zonecfg -z OraEBiz_zone
OraEBiz_zone: No such zone configured
Use 'create' to begin configuring a new zone.
zonecfg:OraEBiz_zone> create
zonecfg:OraEBiz_zone> set zonepath=/export/zones/OraEBiz_zone
```

Step 2: Add the network information to the zone configuration.

```
zonecfg:OraEBiz_zone> add net
zonecfg:OraEBiz_zone:net> set address=10.212.98.193
zonecfg:OraEBiz_zone:net> set physical=bge0
zonecfg:OraEBiz_zone:net> end
```

Step 3: Add a comment for the zone. This is an optional step.

```
zonecfg:OraEBiz_zone> add attr
zonecfg:OraEBiz_zone:attr> set name=comment
zonecfg:OraEBiz_zone:attr> set type=string
zonecfg:OraEBiz_zone:attr> set value="This is
OraEBiz_zone zone for Oracle e-Business Components."
zonecfg:OraEBiz_zone:attr> end
```

Step 4: Verify and commit the zone configuration.

```
zonecfg:OraEBiz_zone> verify
zonecfg:OraEBiz_zone> commit
zonecfg:OraEBiz_zone> exit
bash-3.00# zoneadm list -cv
ID NAME STATUS PATH
0 global running /
- OraEBiz_zone configured /export/zones/OraEBiz_zone
```

Step 5: Install the zone.

```
bash-3.00# zoneadm list -cv
ID NAME STATUS PATH
0 global running /
- OraEBiz_zone configured /export/zones/OraEBiz_zone
```

```
bash-3.00# zoneadm -z OraEBiz_zone install
Preparing to install zone <OraEBiz_zone>.
Creating list of files to copy from the global zone.
Copying <6208> files to the zone.
Initializing zone product registry.
Determining zone package initialization order.
Preparing to initialize <1420> packages on the zone.
Initialized <1420> packages on zone.
Zone <OraEBiz_zone> is initialized.
Installation of <113> packages was skipped.
Installation of these packages generated warnings: <VRTSat>
The file </export/zones/OraEBiz_zone/root/var/sadm/system/
logs/install_log>
contains a log of the zone installation.

bash-3.00# zoneadm list -cv
ID NAME STATUS PATH
0 global running /
- OraEBiz_zone installed /export/zones/OraEBiz_zone
```

Step 7: Configure the zone.

To configure the zone for the first time do the following.
Login to the zone console from the first terminal with
the following command:

```
bash-3.00# zlogin -C OraEBiz_zone
[Connected to zone 'OraEBiz_zone' console]
Now, from the second terminal, start the zone.

bash-3.00# zoneadm -z OraEBiz_zone boot
You will see the following message on the first terminal.
[NOTICE: Zone booting up]
SunOS Release 5.10 Version Generic_118833-36 64-bit
Copyright 1983-2006 Sun Microsystems, Inc. All rights reserved.
Use is subject to license terms.
Hostname: OraEBiz_zone
Loading smf(5) service descriptions: 25/25
Select a Language
0. English
1. Japanese
2. Korean
3. Simplified Chinese
```


4. Traditional Chinese

Please make a choice (0 - 4), or press h or ? for help:

For more information on setting up zones, refer to the *Solaris 10 Administration Guide*.

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

This chapter includes the following topics:

- [Before you install the Cluster Server agent for Oracle e-Business Components](#)
- [About the ACC library](#)
- [Installing the ACC library](#)
- [Installing the agent in a VCS environment](#)
- [Uninstalling the agent in a VCS environment](#)
- [Removing the ACC library](#)
- [Upgrading the agent for OracleApps](#)

Before you install the Cluster Server agent for Oracle e-Business Components

You must install the Cluster Server agent for Oracle e-Business Components on all the systems that will host Oracle e-Business Components service groups.

Before you install the agent for Oracle e-Business Components, ensure that the following prerequisites are met.

- Install and configure Cluster Server.
For more information on installing and configuring Cluster Server, refer to the Cluster Server installation and configuration guides.
- Remove any previous version of this agent.
To remove the agent,
See [“Uninstalling the agent in a VCS environment”](#) on page 33.
- Install the latest version of ACC Library.
To install or update the ACC Library package, locate the library and related documentation in the Agent Pack tarball.
See [“About the ACC library”](#) on page 27.

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 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 the ACC library

The operations of a Cluster Server agent depend on a set of Perl modules known as the ACC library. The library must be installed on each system in the cluster that runs the agent. The ACC library contains common, reusable functions that perform tasks, such as process identification, logging, and system calls.

Instructions to install or remove the ACC library on a single system in the cluster are given in the following sections. The instructions assume that the ACCLib tar file has already been extracted.

Note: The LogDbg attribute should be used to enable debug logs for the ACCLib-based agents when the ACCLib version is 6.2.0.0 or later and VCS version is 6.2 or later.

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 a superuser.
- 2 Download ACC Library.

You can download either the complete Agent Pack tar file or the individual ACCLib tar file from the Veritas Services and Operations Readiness Tools (SORT) site (<https://sort.veritas.com/agents>).

- 3 If you downloaded the complete Agent Pack tar file, navigate to the directory containing the package for the platform running in your environment.

| | |
|---------|--|
| AIX | <code>cd1/aix/vcs/application/acc_library/version_library/pkggs</code> |
| Linux | <code>cd1/linux/generic/vcs/application/acc_library/version_library/rpms</code> |
| Solaris | <code>cd1/solaris/dist_arch/vcs/application/acc_library/version_library/pkggs</code> where <code>dist_arch</code> is <code>sol_sparc</code> . |

- 4 If you downloaded the individual ACCLib tar file, navigate to the `pkggs` directory (for AIX and Solaris), or `rpms` directory (for Linux).
- 5 Install the package. Enter **Yes**, if asked to confirm overwriting of files in the existing package.

| | |
|---------|--|
| AIX | <code># installp -ac -d VRTSaccLib.bff VRTSaccLib</code> |
| Linux | <code># rpm -i \</code> <code>VRTSaccLib-VersionNumber-GA_GENERIC.noarch.rpm</code> |
| Solaris | <code># pkgadd -d VRTSaccLib.pkg</code> |

Note: The LogDbg attribute should be used to enable debug logs for the ACCLib-based agents when the ACCLib version is 6.2.0.0 or later and VCS version is 6.2 or later.

Installing the ACC library IPS package on Oracle Solaris 11 systems

Install the ACC library IPS package on an Oracle Solaris 11 system.

To install the ACC library IPS package on Oracle Solaris 11 systems

- 1 Copy the `VRTSacclib.p5p` package from the `pkgs` directory to the system in the `/tmp/install` directory.
- 2 Disable the publishers that are not reachable as package install may fail, if any, of the already added repositories are unreachable.

```
# pkg set-publisher --disable <publisher name>
```
- 3 Add a file-based repository in the system.

```
# pkg set-publisher -g /tmp/install/VRTSacclib.p5p Veritas
```
- 4 Install the package.

```
# pkg install --accept VRTSacclib
```
- 5 Remove the publisher from the system.

```
# pkg unset-publisher Veritas
```
- 6 Enable the publishers that were disabled earlier.

```
# pkg set-publisher --enable <publisher name>
```

Installing the ACC library package on Solaris brand non-global zones

With Oracle Solaris 11, you must install the ACC library package inside non-global zones. The native non-global zones are called Solaris brand zones.

To install the ACC library package on Solaris brand non-global zones

- 1 Ensure that the SMF services,
`svc:/application/pkg/system-repository:default` and
`svc:/application/pkg/zones-proxyd:default`, are online on the global zone.

```
# svcs svc:/application/pkg/system-repository:default
```

```
# svcs svc:/application/pkg/zones-proxyd:default
```
- 2 Log on to the non-global zone as a superuser.
- 3 Ensure that the SMF service
`svc:/application/pkg/zones-proxy-client:default` is online inside the non-global zone:

```
# svcs svc:/application/pkg/zones-proxy-client:default
```
- 4 Copy the `VRTSacclib.p5p` package from the `pkgs` directory to the non-global zone (for example, at the `/tmp/install` directory).

- 5 Disable the publishers that are not reachable, as package install may fail, if any of the already added repositories are unreachable.

```
# pkg set-publisher --disable <publisher name>
```
- 6 Add a file-based repository in the non-global zone.

```
# pkg set-publisher -g/tmp/install/VRTSacclib.p5p Veritas
```
- 7 Install the package.

```
# pkg install --accept VRTSacclib
```
- 8 Remove the publisher on the non-global zone.

```
# pkg unset-publisher Veritas
```
- 9 Clear the state of the SMF service, as setting the file-based repository causes the SMF service `svc:/application/pkg/system-repository:default` to go into the maintenance state.

```
# svcadm clear svc:/application/pkg/system-repository:default
```
- 10 Enable the publishers that were disabled earlier.

```
# pkg set-publisher --enable <publisher>
```

Note: Perform steps 2 through 10 on each non-global zone.

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 in a VCS environment

- 1 Download the agent from the Veritas Services and Operations Readiness Tools (SORT) site: <https://sort.veritas.com/agents>.

You can download either the complete Agent Pack tar file or an individual agent tar file.
- 2 Uncompress the file to a temporary location, say /tmp.

- 3** If you downloaded the complete Agent Pack tar file, navigate to the directory containing the package for the platform running in your environment.

| | |
|---------|---|
| AIX | <code>cd1/aix/vcs/application/oracleapps_agent/ vcs_version/version_agent/pkg</code> |
| Linux | <code>cd1/linux/generic/vcs/application/oracleapps_agent/ vcs_version/version_agent/rpms</code> |
| Solaris | <code>cd1/solaris/dist_arch/vcs/application/oracleapps_agent/ vcs_version/version_agent/pkg</code> where, <i>dist_arch</i> is <i>sol_sparc</i> |

If you downloaded the individual agent tar file, navigate to the `pkgs` directory (for AIX and Solaris), or `rpms` directory (for Linux).

- 4** Log in as a superuser.
- 5** Install the package.

| | |
|---------|--|
| AIX | <code># installp -ac -d VRTSvcsoa11.rte.bff VRTSvcsoa11.rte</code> |
| Linux | <code># rpm -ihv \ VRTSvcsoa11-AgentVersion-GA_GENERIC.noarch.rpm</code> |
| Solaris | <code># pkgadd -d . VRTSvcsoa11</code> |

After installing the agent package, you must import the agent type configuration file.

See [“Importing the agent types files in a VCS environment”](#) on page 39.

Installing the agent IPS package on Oracle Solaris 11 systems

To install the agent IPS package on an Oracle Solaris 11 system

- 1 Copy the `VRTSvcsoa11.p5p` package from the `pkgs` directory to the system in the `/tmp/install` directory.

- 2 Disable the publishers that are not reachable as package install may fail, if any of the already added repositories are unreachable.

```
# pkg set-publisher --disable <publisher name>
```

where the publisher name is obtained using the `pkg publisher` command.

- 3 Add a file-based repository in the system.

```
# pkg set-publisher -g /tmp/install/VRTSvcsoa11.p5p Veritas
```

- 4 Install the package.

```
# pkg install --accept VRTSvcsoa11
```

- 5 Remove the publisher from the system.

```
# pkg unset-publisher Veritas
```

- 6 Enable the publishers that were disabled earlier.

```
# pkg set-publisher --enable <publisher name>
```

Installing agent packages on Solaris brand non-global zones

To install the agent package on Solaris brand non-global zones

- 1 Ensure that the SMF services,
`svc:/application/pkg/system-repository:default` and
`svc:/application/pkg/zones-proxyd:default`, are online on the global zone.

```
# svcs svc:/application/pkg/system-repository:default
```

```
# svcs svc:/application/pkg/zones-proxyd:default
```

- 2 Log on to the non-global zone as a superuser.

- 3 Ensure that the SMF service
`svc:/application/pkg/zones-proxy-client:default` is online inside non-global zone:

```
# svcs svc:/application/pkg/zones-proxy-client:default
```

- 4 Copy the `VRTSvcsoa11.p5p` package from the `pkgs` directory to the non-global zone (for example, at the `/tmp/install` directory).

- 5 Disable the publishers that are not reachable, as package install may fail, if any of the already added repositories are unreachable.

```
# pkg set-publisher --disable <publisher name>
```
- 6 Add a file-based repository in the non-global zone.

```
# pkg set-publisher -g/tmp/install/VRTSvcsoa11.p5p Veritas
```
- 7 Install the package.

```
# pkg install --accept VRTSvcsoa11
```
- 8 Remove the publisher on the non-global zone.

```
# pkg unset-publisher Veritas
```
- 9 Clear the state of the SMF service, as setting the file-based repository causes the SMF service `svc:/application/pkg/system-repository:default` to go into the maintenance state.

```
# svcadm clear svc:/application/pkg/system-repository:default
```
- 10 Enable the publishers that were disabled earlier.

```
# pkg set-publisher --enable <publisher>
```

Note: Perform steps 2 through 10 on each non-global zone.

Installing the agent in a Solaris 10 brand zone

To install the Oracle e-Business Components agent in a Solaris 10 brand zone:

- Ensure that the ACC library package, `VRTSacclib`, is installed in the non-global zone.

To install `VRTSacclib` in the non-global zone, run the following command from the global zone:

```
# pkgadd -R /zones/zone1/root -d VRTSacclib.pkg
```

- To install the agent package in the non-global zone, run the following command from the global zone:

```
# pkgadd -R zone-root/root -d . VRTSvcsoa11
```

For example: `# pkgadd -R /zones/zone1/root -d . VRTSvcsoa11`

Uninstalling 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 running the following command from any node in the cluster:

```
# haconf -makerw
```

- 3 Remove all Oracle e-Business Components resources from the cluster. Run 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 running 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 running 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.

Run the following command to uninstall the agent:

```
AIX          # installp -u VRTSvcsoa11.rte
```

```
Linux        # rpm -e VRTSvcsoa11
```

```
Solaris      # pkgrm VRTSvcsoa11
```

Note: To uninstall the agent IPS package on a Solaris 11 system, run the following command:

```
# pkg uninstall 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
```

```
Linux        # rpm -e VRTSacclib
```

```
Solaris      # pkgrm VRTSacclib
```

Note: To uninstall the ACCLib IPS package on a Solaris 11 system, run the following command:

```
# pkg uninstall VRTSacclib
```

Upgrading the agent for OracleApps

Perform the following steps to upgrade the agent with minimal disruption.

To upgrade the agent for OracleApps

- 1 Verify the version of the agent for OracleApps.

```
Linux # rpm -qi VRTSvcsoa11 | grep Version
```

```
Solaris # pkginfo -l VRTSvcsoa11 | grep VERSION
```

The output resembles:

```
Version : 6.2.0.0
```

- 2 Save the VCS configuration.

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

3 Identify the OracleApps resource and service group.

```
# hatype -resources OracleApps
```

The output resembles:

```
oracleapps
```

```
# hares -display oracleapps | grep Group
```

The output resembles:

```
oracleapps      Group      global      oa_grp
```

4 Check the type-level attribute—Version.

```
# hatype -display OracleApps | grep Version
```

5 Freeze the OracleApps service group.

```
# hagrps -freeze oa_grp
```

6 Check if the OracleApps agent is running.

```
# haagent -display OracleApps | grep Running
```

The output resembles:

```
OracleApps      Running      Yes
```

7 If the agent is running, stop the agent.

```
# haagent -stop OracleApps -force -sys hostname
```

8 Verify the status of the agent.

```
# haagent -display OracleApps | grep Running
```

The output resembles:

```
OracleApps      Running      No
```

9 Uninstall the agent.

See [“Uninstalling the agent in a VCS environment”](#) on page 33.

10 Install the latest agent.

See [“Installing the agent in a VCS environment”](#) on page 30.

11 Update the agent type definition.

```
# haconf -makerw

# hatype -modify OracleApps ArgList ResLogLevel, State, IState,
User, OracleAppsVersion, OracleHome, ScriptHome, EnvFile, ServerType,
ServerName, WLSPassword, AppsPassword, Port, DBConnectString,
SecondLevelMonitor, MonitorProgram, VirtualHostname

# haattr -add OracleApps ServerName -string

# haattr -add OracleApps WLSPassword -string

# haattr -add OracleApps AppsPassword -string

# haconf -dump -makero
```

12 Start the OracleApps database.

```
# haagent -start OracleApps -sys hostname
```

The output resembles:

```
VCS NOTICE V-16-1-10001 Please look for messages in the log file
```

13 Verify the status of the agent.

```
# haagent -display OracleApps | grep Running
```

The output resembles:

```
OracleApps          Running          Yes
```

14 Unfreeze the OracleApps service group.

```
# hagr -unfreeze oa_grp
```

Configuring the agent for Oracle e-Business Components

This chapter includes the following topics:

- [About configuring the Cluster Server agent for Oracle e-Business Components](#)
- [Importing the agent types files in a VCS environment](#)
- [Agent attributes for Oracle e-Business components](#)
- [Executing a customized monitoring program](#)
- [Creating soft links to JDBC properties file](#)

About configuring the Cluster Server agent for Oracle e-Business Components

After installing the Cluster Server agent for Oracle e-Business Components, you must import the agent type configuration file. After importing this file, review the attributes table that describes the resource type and its attributes, and then create and configure Oracle e-Business Components resources.

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

See [“About sample configurations for the agents for Oracle e-Business Components”](#) on page 60.

Importing the agent types files in a VCS environment

To use the agent for Oracle e-Business Components, you must import the agent types file into the cluster. You can import the agent types file using the VCS graphical user interface or using the command line interface.

To import the agent types file using the VCS Java GUI

- 1 Start the Cluster Manager (Java Console) 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 | ■ AIX | /etc/VRTSvcs/conf/sample_OracleApps/ |
| | ■ Linux | OracleAppsTypes.cf |
| | ■ Solaris | |
| VCS 4.x | Solaris Zones | /etc/VRTSvcs/conf/sample_OracleApps/OracleAppsTypes_zones.cf |
| VCS 5.x or later | ■ AIX | /etc/VRTSagents/ha/conf/OracleApps/ |
| | ■ Linux | OracleAppsTypes.cf |
| VCS 5.0 | Solaris SPARC | /etc/VRTSagents/ha/conf/OracleApps/OracleAppsTypes50.cf |
| VCS 5.1 or later | Solaris SPARC | /etc/VRTSagents/ha/conf/OracleApps/OracleAppsTypes51.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 *Cluster Server Administrator's Guide*.

Agent attributes for Oracle e-Business components

[Table 4-1](#) lists the attributes required for configuring an Oracle component instance.

Table 4-1 Required attributes

| Required attributes | Description |
|---------------------|--|
| ResLogLevel | <p>The logging detail performed by the agent for the resource. Valid values are:</p> <ul style="list-style-type: none"> ■ 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. <p>Type and dimension: string-scalar</p> <p>Default: INFO</p> <p>Example: TRACE</p> <p>Note: You must use the LogDbg attribute instead of the ResLogLevel attribute to enable debug logs for the ACCLib-based agents, when the ACCLib version is 6.2.0.0 or later and the VCS version is 6.2 or later. The agent captures the first failure data of the unexpected events and automatically logs debug messages in their respective agent log files.</p> |
| User | <p>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.</p> <p>Type and dimension: string-scalar</p> <p>Default: ""</p> <p>Example: oracle</p> |

Table 4-1 Required attributes (*continued*)

| Required attributes | Description |
|---------------------|---|
| OracleAppsVersion | <p>The Oracle Applications e-Business version number</p> <p>The valid values are:</p> <ul style="list-style-type: none"> ■ 11: For configuring Oracle Applications e-Business 11i release ■ 12: For configuring Oracle Applications e-Business R12 release ■ 12.2: For configuring Oracle Applications e-Business R12.2 release <p>Type and dimension: integer</p> <p>Default: ""</p> <p>Example: 12</p> |

Table 4-1 Required attributes (*continued*)

| Required attributes | Description |
|---------------------|---|
| ORACLE_HOME | <p>The absolute path to the directory that contains the Oracle client libraries that are located within the Oracle e-Business software directory.</p> <p>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.</p> <p>This attribute is used in the monitor function to locate the environment setup file, and the Oracle client binaries that are used in the second-level monitor routines.</p> <p>For 11i release, this attribute is not required:</p> <ul style="list-style-type: none"> ■ when the SecondLevelMonitor attribute is equal to 0. ■ when the ServerType attribute is equal to FormsMetricsServer, FormsMetricsClient, FormsServer, ReportServer, or WebServer. <p>Type and dimension: string-scalar</p> <p>Default: ""</p> <p>Example for 11i: /ebiz/visora/8.0.6</p> <p>Example for R12 (for 10.1.2 ORACLE_HOME): /ebiz/oracle/VIS/apps/tech_st/10.1.2</p> <p>Example for R12 (for 10.1.3 ORACLE_HOME): /ebiz/oracle/VIS/apps/tech_st/10.1.3</p> <p>Example for R12.2 (for OFM_HOME): /ebiz/d01/fs1/FMW_Home/webtier</p> <p>Example for R12.2 (for 10.1.2 ORACLE_HOME): /ebiz/d01/fs1/EBSapps/10.1.2</p> |
| ScriptHome | <p>The absolute path of the Oracle e-Business scripts directory. This directory contains the scripts to start and stop an e-Business component instance.</p> <p>Type and dimension: string-scalar</p> <p>Default: ""</p> <p>Example for 11i: \$COMMON_TOP/admin/scripts/VIS_ebiz</p> <p>Example for R12 and R12.2: \$INST_TOP/admin/scripts</p> |

Table 4-1 Required attributes (*continued*)

| Required attributes | Description |
|---------------------|---|
| EnvFile | <p>The directory path of the file that must be sourced with the UNIX shell.</p> <ul style="list-style-type: none"> ■ 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. <p>Note: The ORACLE_HOME and ORACLE_SID values must be exported to the environment file.</p> <p>Type and dimension: string-scalar</p> <p>Default: ""</p> <p>Example for 11i: /ebiz/visappl/APPSVIS_ebiz.env</p> <p>Example for R12: /ebiz/oracle/inst/apps/VIS_ebiz1/ora/10.1.3/VIS_ebiz.env</p> |

Table 4-1 Required attributes (*continued*)

| Required attributes | Description |
|---------------------|--|
| ServerType | <p>Type of server.</p> <p>Valid values for Oracle Apps 11i are:</p> <ul style="list-style-type: none"> ■ Listener ■ FormsServer ■ FormsMetricsServer ■ FormsMetricsClient ■ ReportServer ■ WebServer ■ Discoverer ■ FulfillmentServer <p>Valid values for Oracle Apps R12 are:</p> <ul style="list-style-type: none"> ■ OPMN ■ WebServer ■ OAFM ■ OACORE ■ FormsServer ■ Listener ■ FulfillmentServer <p>Valid values for Oracle Apps 12.2 are:</p> <ul style="list-style-type: none"> ■ OPMN ■ WebServer ■ OAFM ■ OACORE ■ FormsServer ■ Listener ■ FulfillmentServer ■ AdminServer ■ NodeManager <p>Type and dimension: string-scalar</p> <p>Default: ""</p> <p>Example: FormsServer</p> |

Table 4-1 Required attributes (*continued*)

| Required attributes | Description |
|---------------------|--|
| AppsPassword | <p>The password of the Apps schema. The value of this attribute must be encrypted using the <code>\$VCS_HOME/bin/vcseencrypt</code> utility.</p> <p>Note: If the <code>ServerType</code> is AdminServer, you must specify this attribute.</p> <p>Type and dimension: string-scalar</p> <p>Default: ""</p> <p>Example: (encrypted)</p> |
| Port | <p>The port number that is associated with a particular type of server, as specified in the <code>ServerType</code> attribute.</p> <p>The monitor function uses this attribute as an argument for second-level monitor check.</p> <p>For Oracle Apps 11i, this attribute is not required when the,</p> <ul style="list-style-type: none"> SecondLevelMonitor attribute is equal to 0. ServerType attribute is equal to FormsMetricsServer, FormsMetricsClient, Discoverer. <p>Note: For Oracle Apps R12, this attribute is required only when the FormServer is configured in the socket mode.</p> <p>For Oracle Apps R12.2, when the server type is OACORE, OAFM, or FormsServer, you must specify the port number of WebLogic AdminServer as the value of this attribute.</p> <p>Type and dimension: integer-scalar</p> <p>Default: ""</p> <p>Example: 8080</p> |
| DBConnectString | <p>The string used to connect database to the listener. This attribute is used while performing second-level monitoring on the listener.</p> <p>Type and dimension: string-scalar</p> <p>Default: ""</p> <p>Example: VIS</p> |

Table 4-1 Required attributes (*continued*)

| Required attributes | Description |
|---------------------|---|
| VirtualHostname | <p>The virtual hostname that is associated with an Oracle e-Business component instance.</p> <p>Type and dimension: string-scalar</p> <p>Default: ""</p> <p>Example: ora_com_01</p> |

[Table 4-2](#) lists the optional attributes for configuring an Oracle component instance.

Table 4-2 Optional attributes

| Optional attributes | Description |
|---------------------|--|
| LogDbg | <p>For ACCLib-based agents, you must use the LogDbg resource type attribute to enable the debug logs when the ACCLib version is 6.2.0.0 or later and the VCS version is 6.2 or later.</p> <p>Set the LogDbg attribute to DBG_5 to enable debug logs for ACCLIB based agent. By default, setting the LogDbg attribute to DBG_5 enables debug logs for all OracleApps resources in the cluster. If debug logs must be enabled for a specific OracleApps resource, override the LogDbg attribute.</p> <p>Type and dimension: string-keylist</p> <p>Default: {} (none)</p> <p>For more information on how to use the LogDbg attribute, refer to the <i>Cluster Server Administrator's Guide</i>.</p> |
| MonitorProgram | <p>The full pathname and command-line arguments for an externally provided monitor program.</p> <p>For information about setting this attribute, See "Executing a customized monitoring program" on page 48.</p> <p>Type and dimension: string-scalar</p> <p>Default: ""</p> <p>Example 1: /usr/ebiz/visappl/work/myMonitor.sh</p> <p>Example 2: /usr/ebiz/visappl/work/myMonitor.sh arg1 arg2</p> |

Table 4-2 Optional attributes (*continued*)

| Optional attributes | Description |
|---------------------|--|
| SecondLevelMonitor | <p>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.</p> <p>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, upper limit is not defined for SecondLevelMonitor.</p> <p>Type and dimension: integer-scalar</p> <p>Default: 0</p> <p>Example: 1</p> |
| ServerName | <p>The name of the Weblogic Server. This attribute is to be used only when the OracleApps version is 12.2 and the server type is OACORE, OAFM, or FormsServer in the servlet mode.</p> <p>Note: This attribute is required only for e-Business Release R12.2.</p> <p>Type and dimension: string-scalar</p> <p>Default: ""</p> <p>Example: oacore_server1</p> |
| WLSPassword | <p>The password for the Weblogic Admin Server. The password must be encrypted using the <code>vcseencrypt</code> command. This attribute must be used only when the OracleApps version is 12.2 and the server type is OACORE, OAFM, FormsServer, AdminServer, or NodeManager in the servlet mode.</p> <p>Note: This attribute is required only for e-Business Release R12.2.</p> <p>Type and dimension: string-scalar</p> <p>Default: ""</p> <p>Example: HTIvKTITNnINjNKnL (encrypted string)</p> |

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, Veritas 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, Veritas 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 a 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 the service groups for Oracle e-Business Components using the CLI

This chapter includes the following topics:

- [About configuring service groups for Oracle e-Business Components](#)
- [Before configuring the service groups for Oracle e-Business Components](#)
- [Configuring Oracle e-Business Components resources for Solaris zones support](#)

About configuring service groups for Oracle e-Business Components

Configuring the Oracle e-Business Components service group involves creating the Oracle e-Business Components service group, its resources, and defining attribute values for the configured resources. You must have administrator privileges to create and configure a service group.

You can configure the service groups using one of the following:

- The Cluster Manager (Java console)
- Veritas Infoscale Operations Manager
- The command line

Before configuring the service groups for Oracle e-Business Components

Before you configure the Oracle e-Business Components service group, you must:

- Verify that the Cluster Server components are installed and configured on all nodes in the cluster where you will configure the service group.
For more information on installing the components, refer to the *InfoScale Availability Installation Guide*.
- Verify that the Cluster Server agent for Oracle e-Business Components is installed on all nodes in the cluster.
See [“Installing the agent in a VCS environment”](#) on page 30.

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.
- Preferably, follow the Veritas 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.
- In a VCS environment, ensure that you have set the value of ContainerName attribute to the name of the Solaris zone.
By default, the agent function executes in the Global zone.

Troubleshooting the agent for Oracle e-Business Components

This chapter includes the following topics:

- [Using the 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 the correct software and operating system versions

Ensure that you use correct software and operating system versions.

For information on the software versions that the agent for Oracle e-Business Components supports, see the Veritas Services and Operations Readiness Tools (SORT) site: <https://sort.veritas.com/agents>.

Meeting prerequisites

Before installing the agent for Oracle e-Business Components, ensure that the following prerequisites are met.

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

See [“Before you install the Cluster Server agent for Oracle e-Business Components”](#) on page 26.

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 agents 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 Oracle e-Business Components instance outside the cluster framework, is described 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 | f60srvrn, 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* |

| Oracle component for Oracle Apps R12.2 | Oracle script to start or stop the component | List of processes |
|---|---|--------------------------|
| OPMN | adopmnctl.sh | opmnsh |
| Forms Server (Servlet Mode) | admanagedsrvctl.sh | ServerName |

| | | |
|----------------------------|--------------------|--------------------------|
| Forms Server (Socket Mode) | adformsrvctl.sh | frmsrv |
| OACORE Server | admanagedsrvctl.sh | ServerName |
| OAFM Server | admanagedsrvctl.sh | ServerName |
| Web Server | adapcctl.sh | httpd |
| RPC Listener | adalnctl.sh | tnslsnr |
| Fulfillment Server | jtfmctl.sh | java.*FulfillmentServer* |
| AdminServer | adadminsrvctl.sh | AdminServer |
| NodeManager | adnodemgrctl.sh | weblogic.NodeManager |

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

For Oracle Apps 11i

```
# adfrmctl.sh start
```

For Oracle Apps R12

```
# adformsctl.sh start
```

For Oracle Apps R12.2

```
# admanagedsrvctl.sh start forms_server1
```

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.

Note: Ensure that you start the Weblogic administration server before starting the components controlled by the Weblogic server, such as oafm, oacore, and forms. The administration server can started using the `adadminsrvctl.sh` script available in the `ScriptHome` directory.

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 the following command:

For Oracle Apps 11i

```
# adfrmctl.sh stop
```

For Oracle Apps R12

```
# adformsctl.sh stop
```

For Oracle Apps R12.2

```
# admanagedsrvctl.sh stop forms_server1
```

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. Veritas 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`.

The `LogDbg` attribute should be used to enable the debug logs for the ACCLib-based agents when the ACCLIB version is 6.2.0.0 or later and the VCS version is 6.2 or later.

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 Save the configuration changes.

```
# haconf -dump
```

- 9 Review the contents of the log file.

Use the time noted in the previous steps to diagnose the problem.

You can also contact Veritas support for more help.

To enable debug logs for all resources of type OracleApps

- ◆ Enable the debug log.

```
# hatype -modify OracleApps LogDbg DBG_5
```

To override the LogDbg attribute at resource level

- ◆ Override the LogDbg attribute at the resource level and enable the debug logs for the specific resource.

```
# hares -override OracleApps LogDbg
# hares -modify OracleApps LogDbg DBG_5
```

Sample Configurations

This appendix includes the following topics:

- [About sample configurations for the agents for Oracle e-Business Components](#)
- [Sample configuration file for Oracle e-Business Concurrent Manager for Linux](#)
- [Sample agent type definition](#)
- [Sample agent type definitions with Solaris zone support](#)
- [Sample Oracle Application Server resource configuration](#)
- [Sample service group configuration](#)

About sample configurations for the agents 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 agents for Oracle e-Business Components. For more information about these resource types, refer to the *Cluster Server Bundled Agents Reference Guide*.

Sample configuration file for Oracle e-Business Concurrent Manager for Linux

A sample `main.cf` file for Oracle e-Business Concurrent Manager for Linux is as follows:

```
group EBS_CM (
    SystemList = { vmrac020 = 0, vmrac021 = 1 }
```

```

AutoStartList = { vmrac020, vmrac021 }
)

Netlsnr cm_lsnr (
    Critical = 0
    Owner = oracle
    Home = "/ebsmnt/oracle/VIS/12.1.0"
    TnsAdmin = "/ebsmnt/oracle/VIS/12.1.0/network/admin"
    Listener = VIS
)

Oracle cm_db (
    Critical = 0
    Sid = VIS
    Owner = oracle
    Home = "/ebsmnt/oracle/VIS/12.1.0"
    DBName = VIS
)

OracleAppsCM ora_cm (
    Critical = 0
    ResLogLevel = TRACE
    User = oracle
    OracleHome = "/ebsmnt/oracle/VIS/fs1/EBSapps/10.1.2"
    ScriptHome = "/ebsmnt/oracle/VIS/fs1/inst/apps/VIS_ebsora/admin/scripts"
    SQLDIR = "/ebsmnt/oracle/VIS/fs1/EBSapps/appl/fnd/12.0.0/sql"
    EnvFile = "/ebsmnt/oracle/VIS/fs1/EBSapps/appl/VIS_ebsora.env"
    DB_USER = apps
    DB_Password = GSHuGUgUJ
    VirtualHostname = "10.209.69.199"
    ProcMon = { FNDLIBR, FNDMSM }
)

requires group EBS_sg online local soft
cm_db requires cm_lsnr
ora_cm requires cm_db

// resource dependency tree
//
//      group EBS_CM
//      {
//      OracleAppsCM ora_cm
//      {

```

```

//          Oracle cm_db
//          {
//          Netlsnr cm_lsnr
//          }
//          }
//          }

```

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 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 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 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 R12.2

```
OracleApps OraApps_FormsSrvr
(
  User = oracle
  OracleAppsVersion = "12.2"
  OracleHome = "/ebs_mount/ebs/d01/fs1/FMW_Home/webtier"
  ScriptHome = "/ebs_mnt/ebs/d01/fs1/inst/apps/VIS_vc$sx075/admin/scripts"
  ServerType = FormsServer
  ServerName = forms_server1
  WLSPassword = drkPipFppPspKpmPgjEj$Jg
  Port = 7001
)
```

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

For Oracle Apps R12.2, the dependency of resources is as follows:

- OraApps_FormsSrvr requires WebLogic_Admin
- OraApps_OACORE requires WebLogic_Admin
- OraApps_OAFM requires WebLogic_Admin
- OraApps_HTTPSrvr requires OraApps_OPMN
- OraApps_OPMN requires OraApps_ip
- OraApps_OPMN requires OraApps_mnt

- OraApps_mnt requires OraApps_dg
- OraApps_Admin requires OraApps_nm
- OraApps_nm requires OraApps_ip
- OraApps_nm 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, DBConnectionString, SecondLevelMonitor,  
    MonitorProgram, VirtualHostname}  
    str ResLogLevel = INFO  
    str User  
    int OracleAppsVersion  
    str OracleHome  
    str ScriptHome  
    str EnvFile  
    str ServerType  
    str Port  
    str DBConnectionString  
    int SecondLevelMonitor = 0  
    str MonitorProgram  
    str VirtualHostname  
    str ContainerName  
)
```

Sample Oracle Application Server resource configuration

A sample excerpt from the main.cf file is shown below for Oracle e-Business R12.2 resource configuration.

```
include "OracleASMTypes.cf"  
include "types.cf"  
include "Db2udbTypes.cf"  
include "OracleTypes.cf"  
include "OracleAppsTypes51.cf"  
include "SybaseTypes.cf"  
include "WebLogicTypes51.cf"  
  
cluster oraebizcluster (  
    UserNames = { admin = chiAhcHeiDiiGqiChf }  
    Administrators = { admin }  
)  
  
system vcssx173 (  
)  
  
system vcssx174 (  
)
```

```
group EBS_sg (
    SystemList = { vcssx174 = 0, vcssx173 = 1 }
)

IP ip_res (
    Device = e1000g2
    Address = "110.120.62.18"
    NetMask = "255.255.255.0"
)

OracleApps ebs_forms (
    Critical = 0
    ResLogLevel = TRACE
    User = oracle
    OracleAppsVersion = "12.2"
    OracleHome = "/ebs_mnt/ebs/d01/oracle/VIS/
                  fs1/FMW_Home/webtier"
    ScriptHome = "/ebs_mnt/ebs/d01/oracle/VIS/
                  fs1/inst/apps/VIS_vcssx174/admin/scripts"
    ServerType = FormsServer
    ServerName = forms_server1
    WLSPassword = IWOUlWlOJoKOlOMoN
    Port = 7001
    VirtualHostname = "110.120.62.18"
)

OracleApps ebs_ful (
    Critical = 0
    ResLogLevel = TRACE
    User = oracle
    OracleAppsVersion = "12.2"
    OracleHome = "/ebs_mnt/ebs/d01/oracle/VIS/
                  fs1/EBSapps/10.1.2"
    ScriptHome = "/ebs_mnt/ebs/d01/oracle/VIS/
                  fs1/inst/apps/VIS_vcssx174/admin/scripts"
    ServerType = FulfillmentServer
    VirtualHostname = "110.120.62.18"
)

OracleApps ebs_lsnr (
    Critical = 0
    ResLogLevel = TRACE
    User = oracle
```

```
OracleAppsVersion = "12.2"
OracleHome = "/ebs_mnt/ebs/d01/oracle/VIS/
             fs1/EBSapps/10.1.2"
ScriptHome = "/ebs_mnt/ebs/d01/oracle/VIS/
             fs1/inst/apps/VIS_vcssx174/admin/scripts"
ServerType = Listener
WLSPassword = IWOUWLWlOJoKOlOMoN
VirtualHostname = "110.120.62.18"
)

OracleApps ebs_oacore (
    Critical = 0
    ResLogLevel = TRACE
    User = oracle
    OracleAppsVersion = "12.2"
    OracleHome = "/ebs_mnt/ebs/d01/oracle/VIS/
                 fs1/FMW_Home/webtier"
    ScriptHome = "/ebs_mnt/ebs/d01/oracle/VIS/
                 fs1/inst/apps/VIS_vcssx174/admin/scripts"
    ServerType = OACORE
    ServerName = oacore_server1
    WLSPassword = IWOUWLWlOJoKOlOMoN
    Port = 7001
    VirtualHostname = "110.120.62.18"
)

OracleApps ebs_oafm (
    Critical = 0
    ResLogLevel = TRACE
    User = oracle
    OracleAppsVersion = "12.2"
    OracleHome = "/ebs_mnt/ebs/d01/oracle/VIS/
                 fs1/FMW_Home/webtier"
    ScriptHome = "/ebs_mnt/ebs/d01/oracle/VIS/
                 fs1/inst/apps/VIS_vcssx174/admin/scripts"
    ServerType = OAFM
    ServerName = oafm_server1
    WLSPassword = IWOUWLWlOJoKOlOMoN
    Port = 7001
    VirtualHostname = "110.120.62.18"
)

OracleApps ebs_opmn (
```

```
Critical = 0
ResLogLevel = TRACE
User = oracle
OracleAppsVersion = "12.2"
OracleHome = "/ebs_mnt/ebs/d01/oracle/VIS/
             fs1/FMW_Home/webtier"
ScriptHome = "/ebs_mnt/ebs/d01/oracle/VIS/
             fs1/inst/apps/VIS_vcssx174/admin/scripts"
ServerType = OPMN
WLSPassword = IWOUWLWlOJoKOlOMoN
VirtualHostname = "110.120.62.18"
)

OracleApps ebs_web (
    Critical = 0
    ResLogLevel = TRACE
    User = oracle
    OracleAppsVersion = "12.2"
    OracleHome = "/ebs_mnt/ebs/d01/oracle/VIS/
                 fs1/FMW_Home/webtier"
    ScriptHome = "/ebs_mnt/ebs/d01/oracle/VIS/
                 fs1/inst/apps/VIS_vcssx174/admin/scripts"
    ServerType = WebServer
    VirtualHostname = "110.120.62.18"
)

OracleApps ebs_Admin (
    Critical = 0
    ResLogLevel = TRACE
    User = oracle
    OracleAppsVersion = "12.2"
    OracleHome = "/ebs_mnt/ebs/d01/oracle/VIS/
                 fs1/FMW_Home/webtier"
    ScriptHome = "/ebs_mnt/ebs/d01/oracle/VIS/
                 fs1/inst/apps/VIS_vcssx174/admin/scripts"
    ServerType = AdminServer
    WLSPassword = IWOUWLWlOJoKOlOMoN
    AppsPassword = IWOUWLWlOJoKOlOMoN
    Port = 7001
    VirtualHostname = "110.120.62.18"
)

OracleApps ebs_nm (
```

```

Critical = 0
User = oracle
OracleAppsVersion = "12.2"
OracleHome = "/ebs_mnt/oracle/d01/oracle/VIS/fs1/FMW_Home/web
ScriptHome = "/ebs_mnt/oracle/d01/oracle/VIS/fs1/inst/apps/VI
ServerType = NodeManager
WLSPassword = HV0tMTjTTtWToTQtKNiNjNk
VirtualHostname = "110.120.62.18"
)

ebs_Admin requires ebs_nm
ebs_nm requires ip_res
ebs_forms requires admin_res
ebs_oacore requires admin_res
ebs_oafm requires admin_res

// resource dependency tree
//
//      group EBS_sg
//      {
//      OracleApps ebs_forms
//      {
//      OracleApps ebs_Admin
//      {
//      IP ip_res
//      }
//      }
//      OracleApps ebs_ful
//      OracleApps ebs_lsnr
//      OracleApps ebs_oacore
//      {
//      OracleApps ebs_Admin
//      {
//      OracleApps ebs_nm
//      {
//      IP ip_res
//      }
//      }
//      }
//      OracleApps ebs_oafm
//      {
//      OracleApps ebs_Admin

```



```
//      {
//      IP ip_res
//      }
//      }
//      OracleApps ebs_opmn
//      OracleApps ebs_web
//      }

group Storage_sg (
    SystemList = { vcssx174 = 0, vcssx173 = 1 }
)

DiskGroup dg_res (
    Critical = 0
    DiskGroup = oraebiz1_dg
)

Mount mount_res (
    Critical = 0
    FsckOpt = "-y"
    BlockDevice = "/dev/vx/dsk/oraebiz1_dg/ebs_vol"
    MountPoint = "/ebs_mnt/"
    FSType = vxfs
)

Volume vol_res (
    Critical = 0
    Volume = ebs_vol
    DiskGroup = oraebiz1_dg
)

mount_res requires vol_res
vol_res requires dg_res

// resource dependency tree
//
//      group Storage_sg
//      {
//      Mount mount_res
```

```
//      {
//      Volume vol_res
//      {
//      DiskGroup dg_res
//      }
//      }
//      }
```

```
group OraAS10gInfrastructure (
  SystemList = { sysA = 0, sysB = 1 }
)

DiskGroup OraAS10g_dg (
  DiskGroup = oradg
)

IP OraAS10g_IP (
  Device = bge0
  Address = "110.120.62.18"
  NetMask = "255.255.255.0"
)

NIC OraAS10g_NIC (
  Device = bge0
)

Mount OraAS10g_mnt (
  MountPoint = "/ora10g"
  BlockDevice = "/dev/vx/dsk/oradg/oravol"
  FSType = vxfs
  MountOpt = rw
  FsckOpt = "-y"
)

OracleAS9 OraAS10g_HTTPSrvr (
  User = oracle
  OracleHome = "/ora10g/OraHome_1"
  ComponentProcessType = HTTP_Server
)

OracleAS9 OraAS10g_OC4JSecurity (
  Enabled = 0
  User = oracle
```

```
OracleHome = "/ora10g/OraHome_1"
ComponentProcessType = OC4J_SECURITY
)

OracleAS9 OraAS10g_OID (
  Enabled = 0
  User = oracle
  OracleHome = "/ora10g/OraHome_1"
  ComponentProcessType = OID
)

OracleAS9 OraAS10g_OPMN (
  Critical = 0
  User = oracle
  OracleHome = "/ora10g/OraHome_1"
  ComponentProcessType = OPMN
  SecondLevelMonitor = 5
)

OracleAS9 OraAS10g_dbconsole (
  User = oracle
  OracleHome = "/ora10g/OraHome_1"
  ComponentProcessType = dbconsole
)

OracleAS9 OraAS10g_dcm (
  Enabled = 0
  User = oracle
  OracleHome = "/ora10g/OraHome_1"
  ComponentProcessType = dcm-daemon
)

OracleAS9 OraAS10g_iasconsole (
  User = oracle
  OracleHome = "/ora10g/OraHome_1"
  ComponentProcessType = iasconsole
)

OracleAS9 OraAS10g_ReportsServerComponent (
  ResLogLevel = TRACE
  User = oracle
  OracleHome = "/ora10g/OraHome_1"
  ComponentProcessType = ReportsServerComponent
```

```
)  
  
OraAS10g_IP requires OraAS10g_NIC  
OraAS10g_mnt requires OraAS10g_dg  
OraAS10g_HTTPSrvr requires OraAS10g_OPMN  
OraAS10g_OPMN requires OraAS10g_mnt  
OraAS10g_OC4JSecurity requires OraAS10g_OID  
OraAS10g_OID requires OraAS10g_OPMN  
OraAS10g_OPMN requires OraAS10g_IP  
OraAS10g_dbconsole requires OraAS10g_IP  
OraAS10g_dcm requires OraAS10g_OID  
OraAS10g_iasconsole requires OraAS10g_OPMN  
OraAS10g_ReportsServerComponent requires OraAS10g_OPMN
```

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.

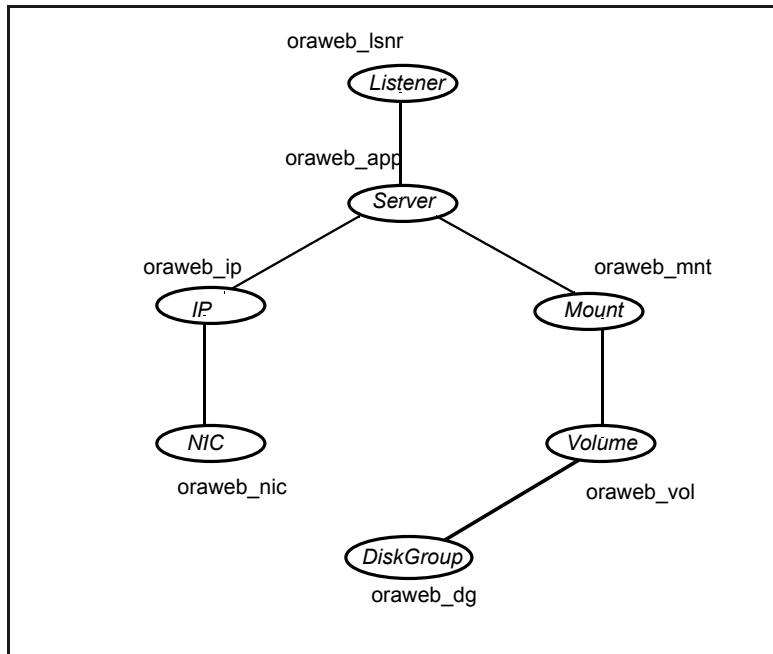
Veritas recommends that a service group must be configured using a virtual IP address and shared disk storage. All resources in a service group share the same virtual IP address, disk storage, and UNIX User account. In event of a failure, an application resource can run on any node in the cluster without 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 illustrates 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](#) illustrates 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

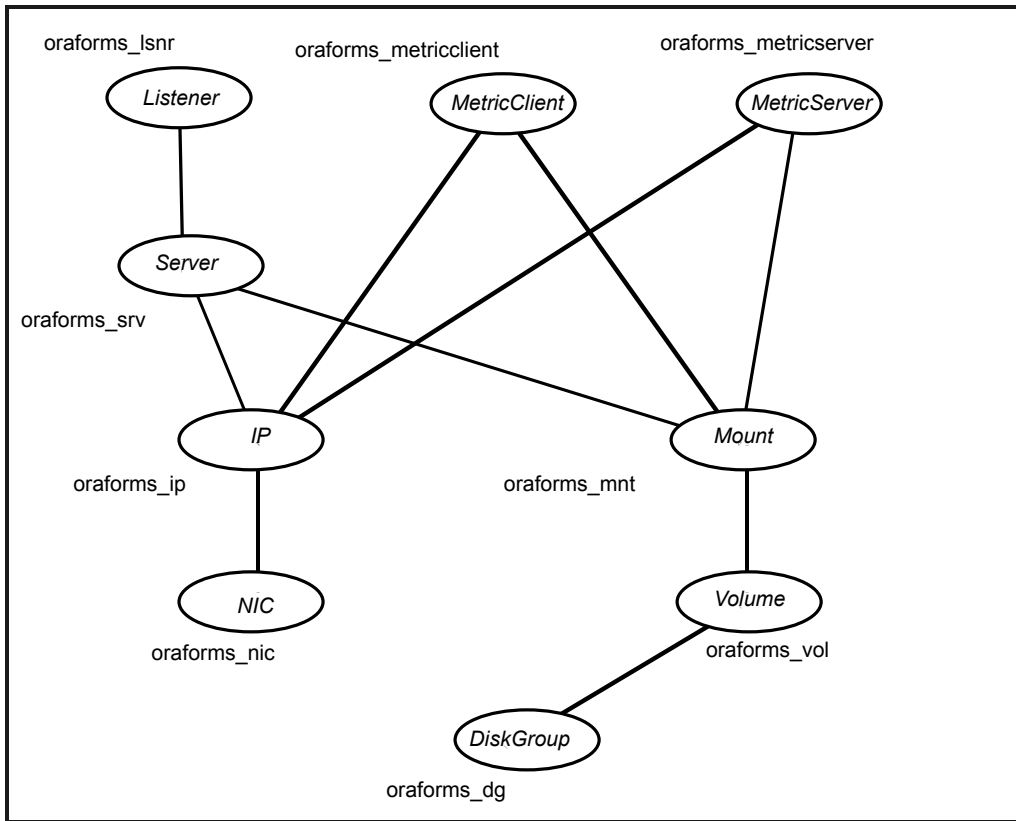


Figure A-3 illustrates a common setup for the Oracle Report Server as a companion to the Oracle Concurrent Manager.

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

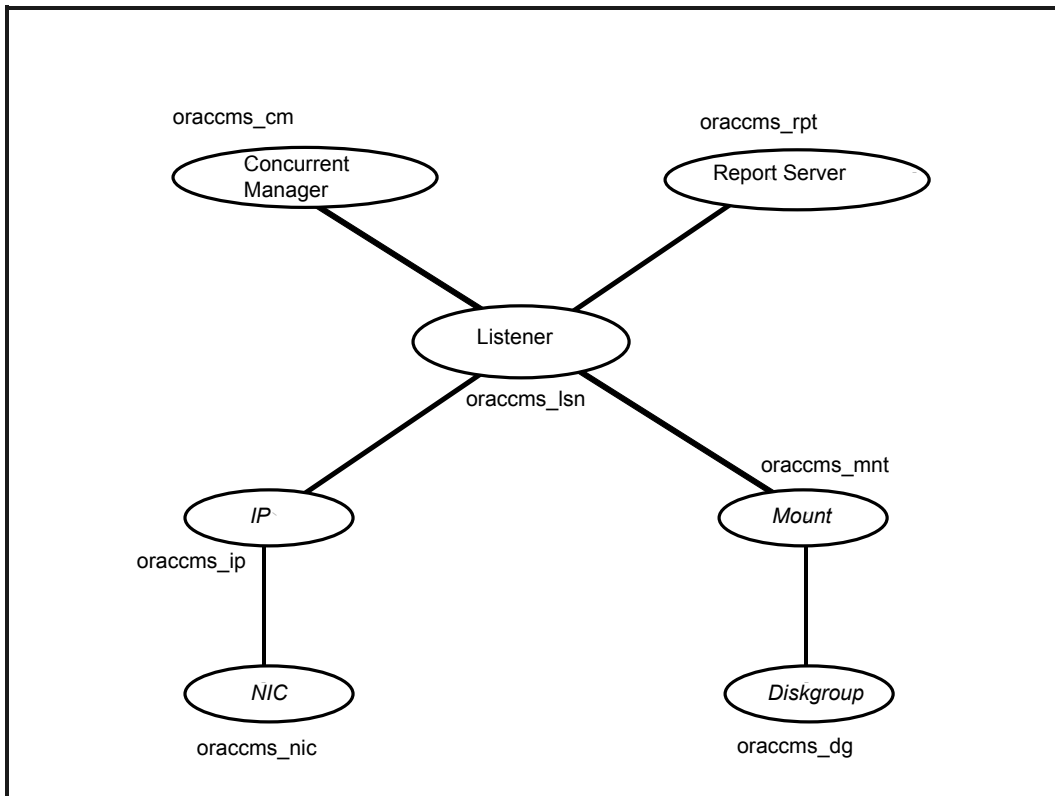
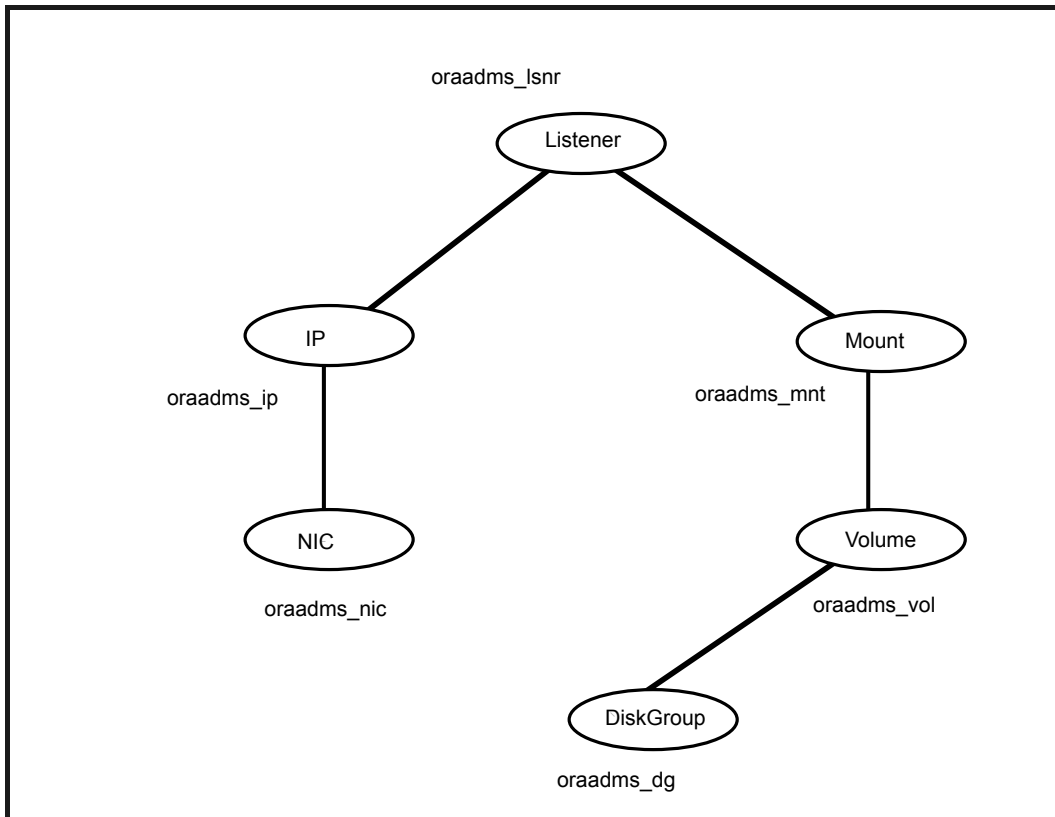


Figure A-4 illustrates a setup for the Oracle Administration server service group.

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](#) illustrates 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

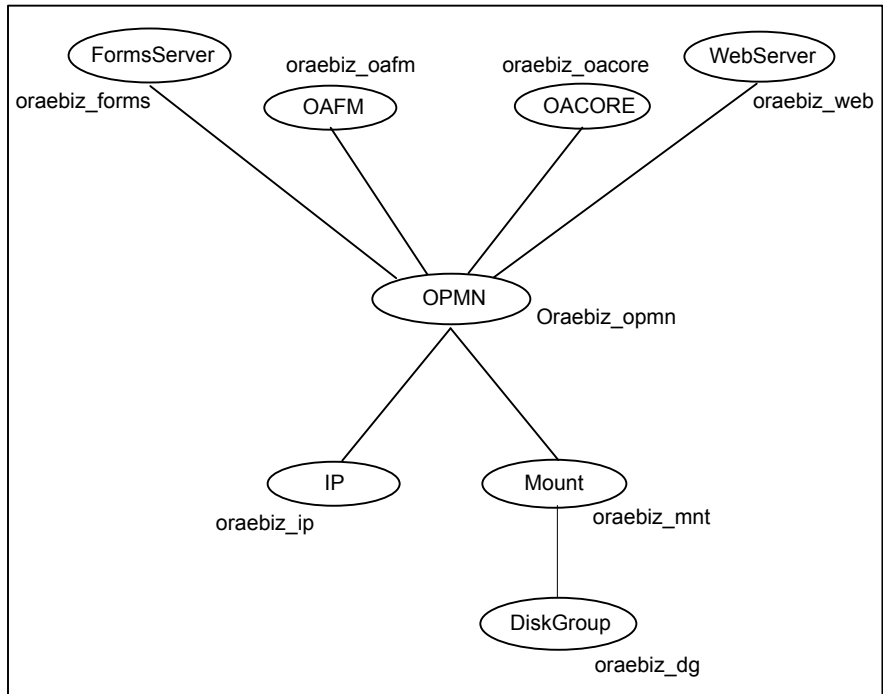


Figure A-6 illustrates a service group with Oracle oAS 10g OPMN and its managed processes; Web Server, Forms, OAFM and OACORE, under Solaris zone.

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

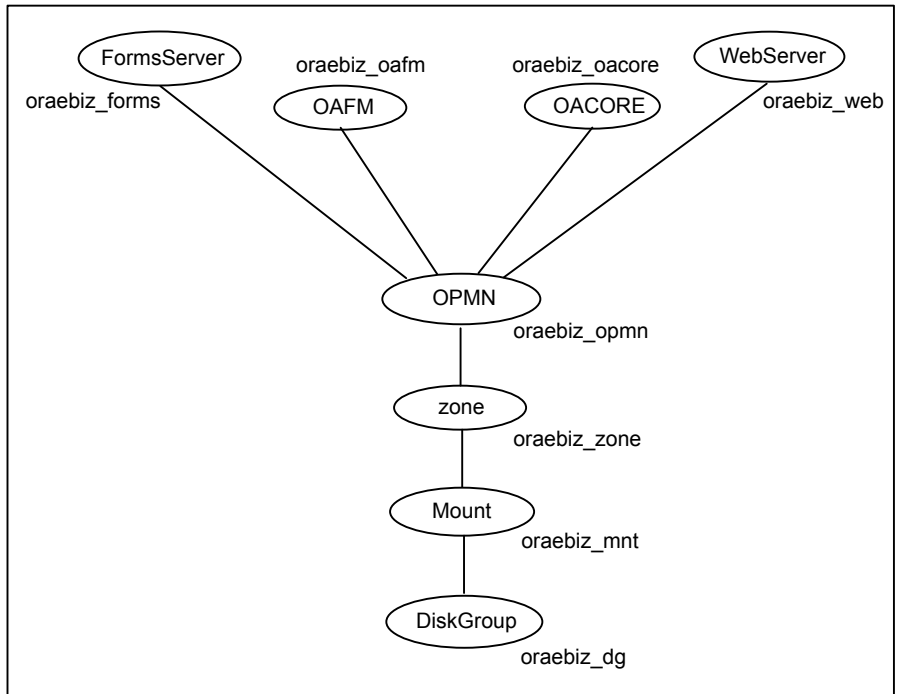
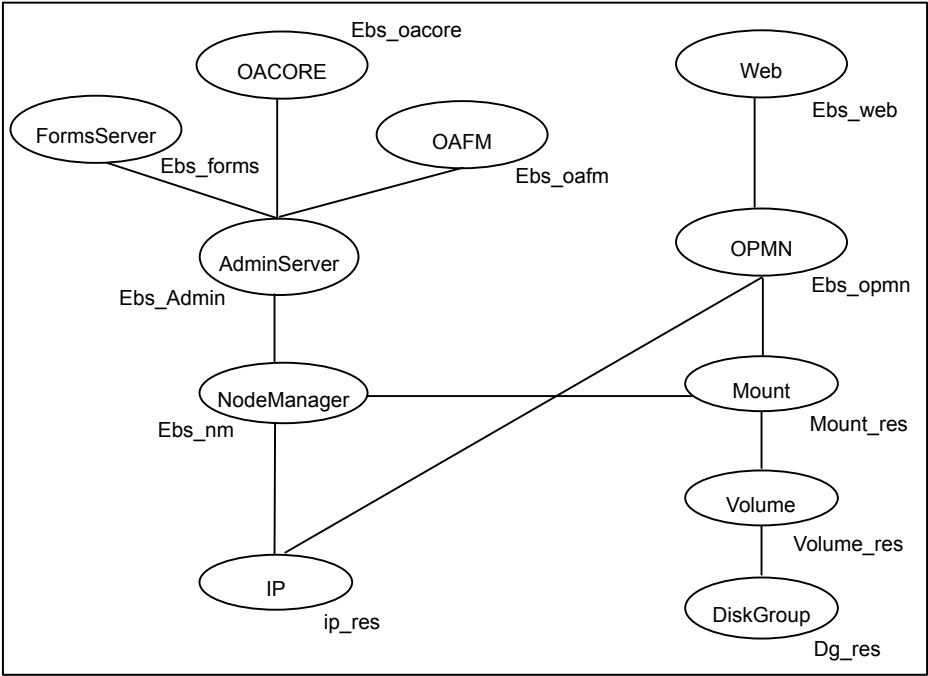


Figure A-7 depicts sample configuration service groups for different components of Oracle e-Business 12.2.

Figure A-7 Sample configuration service groups for Oracle e-Business 12.2 components



Index

A

- About
 - Oracle e-Business Components 16
- about
 - configuring Oracle e-Business Components for high availability 17
 - configuring service groups 50
- about ACC library 27
- ACC library
 - installing 28
 - removing 35
- agent
 - detecting application failure 11
 - importing agent types files 39
 - installing, VCS environment 30
 - overview 7
 - sample service group configuration 76
 - uninstalling, VCS environment 33
- agent attributes 41
 - DBConnectionString 45
 - EnvFile 43
 - MonitorProgram 46
 - ORACLE_HOME 42
 - Port 45
 - ResLogLevel 40
 - ScriptHome 42
 - SecondLevelMonitor 47
 - ServerType 44
 - User 40
 - VirtualHostname 46–47
- agent configuration file
 - importing 39
- agent functions 10
 - clean 11
 - monitor 11
 - offline 10
 - online 10
- agent installation
 - general requirements 26
 - requirements for Solaris zones 27
 - steps to install 30

agent type definition

- Sample Oracle Forms Listener instance 65
- Sample Oracle Forms Server instance 65
- Sample Oracle HTTP Server instance 63
- Sample Oracle OACORE instance 64
- Sample Oracle OAFM instance 64
- Sample Oracle OPMN instance 63

B

- before
 - configuring the service groups 51

C

- configuring monitor function 48

E

- executing custom monitor program 48

L

- logs
 - reviewing agent log files 58
 - reviewing cluster log files 58
 - reviewing error log files 57
 - using trace level logging 58

O

- Oracle e-Business Components
 - configuring resources 53
 - configuring resources for Solaris zones 51
 - starting instance outside cluster 53

S

- sample agent type definitions
 - Ora E-Biz Components, Solaris zone support 67
- Setting
 - Solaris zones 23
- setting
 - Oracle e-Business Components in a cluster 15

- Solaris zone support
 - configuring Oracle e-Business Componentsresources 51
 - installation requirements 27
 - sample agent type definitions 67
- starting the Oracle e-Business Components instance
 - outside a cluster 53

T

- troubleshooting
 - meeting prerequisites 52
 - reviewing error log files 57
 - reviewing agent log files 58
 - reviewing cluster log files 58
 - using trace level logging 58
 - using correct software 52

U

- uninstalling agent, VCS environment 33