

Veritas™ High Availability Agent for Oracle Essbase Server Installation and Configuration Guide

AIX, HP-UX, Linux, Solaris

5.1

Veritas High Availability Agent for Essbase Server Installation and Configuration Guide

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

Agent Version: 5.1.0.0

Document version: 5.1.0.0.1

Legal Notice

Copyright © 2011 Symantec Corporation. All rights reserved.

Symantec, the Symantec logo, Veritas, Veritas Storage Foundation, CommandCentral, NetBackup, Enterprise Vault, and LiveUpdate are trademarks or registered trademarks of Symantec corporation or its affiliates in the U.S. and other countries. Other names may be trademarks of their respective owners.

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 Symantec Corporation 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. SYMANTEC CORPORATION 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, "Rights in Commercial Computer Software or Commercial Computer Software Documentation", as applicable, and any successor regulations. 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.

Symantec Corporation
350 Ellis Street
Mountain View, CA 94043
<http://www.symantec.com>

Technical Support

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

Symantec's support offerings include the following:

- A range of support options that give you the flexibility to select the right amount of service for any size organization
- Telephone and/or Web-based support that provides rapid response and up-to-the-minute information
- Upgrade assurance that delivers software upgrades
- Global support purchased on a regional business hours or 24 hours a day, 7 days a week basis
- Premium service offerings that include Account Management Services

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

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

All support services will be delivered in accordance with your support agreement and the then-current enterprise technical support policy.

Contacting Technical Support

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

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

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

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

- Product release level

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

Licensing and registration

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

www.symantec.com/business/support/

Customer service

Customer service information is available at the following URL:

www.symantec.com/business/support/

Customer Service is available to assist with non-technical questions, such as the following types of issues:

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

Support agreement resources

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

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

Documentation

Product guides are available on the media in PDF format. Make sure that you are using the current version of the documentation. The document version appears on page 2 of each guide. The latest product documentation is available on the Symantec Web site.

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

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

doc_feedback@symantec.com

About Symantec Connect

Symantec Connect is the peer-to-peer technical community site for Symantec's enterprise customers. Participants can connect and share information with other product users, including creating forum posts, articles, videos, downloads, blogs and suggesting ideas, as well as interact with Symantec product teams and Technical Support. Content is rated by the community, and members receive reward points for their contributions.

<http://www.symantec.com/connect/storage-management>

Contents

Technical Support	4
Chapter 1 Introducing the Veritas High Availability Agent for Essbase Server	9
About the Veritas agent for Essbase Server	9
What's new in this agent	10
Supported software	10
How the agent makes Essbase Server highly available	10
High availability for Essbase Server instances running in Solaris zones	11
Essbase server agent functions	11
Online	11
Offline	12
Monitor	12
Clean	12
Chapter 2 Installing and configuring Essbase Server for high availability	15
About Essbase Server	15
Administration Services	16
Essbase Studio	16
Integration Services	17
Provider Services	17
Uniquely identifying Essbase Server server instances	17
About installing Essbase Server for high availability	17
Configuring the Essbase Server for high availability	18
Synchronizing accounts and services	18
Removing physical host dependencies	18
Chapter 3 Installing, upgrading, and removing the agent for Essbase Server	19
Before you install the Veritas agent for Essbase Server	19
About the ACC library	20
Installing the ACC library	20

	Installing the agent in a VCS environment	21
	Removing the agent in a VCS environment	22
	Removing the ACC library	23
Chapter 4	Configuring the agent for Essbase Server	25
	About configuring the Veritas agent for Essbase Server	25
	Importing the agent types files in a VCS environment	25
	Essbase server agent attributes	27
	Executing a customized monitoring program	31
Chapter 5	Configuring the service groups for Essbase Server	33
	About configuring service groups for Essbase Server	33
	Before configuring the service groups for Essbase Server	33
	Configuring service groups for Essbase Server	34
	Generating the environments file for Essbase Server	36
Chapter 6	Troubleshooting the agent for Essbase Server	39
	Using the correct software and operating system versions	39
	Meeting prerequisites	39
	Configuring Essbase resources	40
	Starting the Essbase instance outside a cluster	40
	Reviewing error log files	41
	Reviewing cluster log files	41
	Using trace level logging	41
Appendix A	Sample Configurations	43
	About sample configurations for the agent for Essbase Server	43
	Sample agent type definition	43
	Sample configuration	45
	Sample configuration with Solaris zone support	47
	Sample service group configuration	49
	Sample service group configuration with Solaris zone support	50
Index	53

Introducing the Veritas High Availability Agent for Essbase Server

This chapter includes the following topics:

- [About the Veritas agent for Essbase Server](#)
- [What's new in this agent](#)
- [Supported software](#)
- [How the agent makes Essbase Server highly available](#)
- [Essbase server agent functions](#)

About the Veritas agent for Essbase Server

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

The Veritas agent for Essbase provides high availability for an Essbase server in a cluster. The Veritas High Availability agent brings specific Essbase server instance online, monitor the instance, and brings the instance offline. The Veritas High Availability agent monitors the processes of Essbase server instance and shutdown the Essbase server in case of failure.

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

<http://www.symantec.com/business/support/index?page=content&id=TECH46455>

What's new in this agent

The enhancements in this release of the Essbase agent are as follows:

- Added support for AIX 7.1.

Supported software

The Veritas agent for Essbase server supports the following software versions:

Veritas Cluster Server	<ul style="list-style-type: none">■ AIX—VCS 4.0, 5.0, 5.1■ HP-UX—VCS 4.1, 5.0■ Linux—VCS 4.0, 4.1, 5.0, 5.1■ Solaris—VCS 4.1, 5.0, 5.1 and all intermediate Maintenance Packs of these major releases.
ACC Library	5.2 and later
Operating Systems	<ul style="list-style-type: none">■ AIX 5.2, 6.1, 7.1 on pSeries■ HP UX 11iv2 on PA-RISC and HP-UX 11iv2 and 11iv3 on Itanium■ Red Hat Enterprise Linux 4, 5 on Intel■ Solaris 10 on SPARC
Oracle Essbase Server	11.1.1

How the agent makes Essbase Server highly available

The Veritas agent for Essbase Server continuously monitors the Essbase processes to verify that they function properly.

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 Essbase processes are present in the process table. Process check cannot detect whether processes are in hung or stopped states.
- Secondary or Detail monitoring
In this mode, the agent runs a utility to verify the status of Essbase. The agent detects application failure if the monitoring routine reports an improper

function of the Essbase processes. When this application failure occurs, the Essbase service group fails over to another node in the cluster. Thus, the agent ensures high availability for Essbase.

High availability for Essbase Server 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, *Veritas Cluster Server User's Guide*.

The Veritas agent for Essbase Server is zone-aware and can monitor Essbase instances running in non-global zones.

Essbase server agent functions

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

Online

The online function performs the following tasks:

- Performs the preliminary check to ensure that the Essbase server is not online on the specified node in the cluster.
- Uses Essbase server binaries to start the Essbase server using the `ESSBASE` command available at the bin directory of `ArborPath` attribute value. The online function also sources a shell script or a program that the `EnvFile` attribute specifies. The script or program ensures that the required shell environment variables are properly set before executing the start script.
- Ensures that the Essbase server instance is up and running successfully. The operation uses the wait period that the `OnlineTimeout` attribute specifies, to enable the Essbase server instance to initialize fully before allowing the monitor function to probe the resource.

Offline

The offline function performs the following tasks:

- Verifies that the Essbase server instance is not already offline.
- Uses the Essbase server binaries to stop the Essbase server instance using the `ESSCMD` command available at bin directory of `ArborPath` attribute value. The offline function also sources a shell script or a program that the `EnvFile` attribute specifies. The script or program ensures that the required shell environment variables are properly set before executing the stop script.
- Ensures that the Essbase server is given enough time to go offline successfully. The operation uses a wait period that the `OfflineTimeout` attribute specifies, to allow the Essbase server instance to complete the offline sequence before allowing further probing of the resource.

Monitor

The monitor function monitors the states of the Essbase server instance running on all nodes within the cluster.

The function performs the following task:

- The first level check scans the system process table and searches the processes that must be running for the Essbase server instance. If the first level check does not find these processes running on the node, the check exits immediately, and reports the Essbase server instance as offline.
- If the `SecondLevelMonitor` attribute is set to greater than 0, the monitor function performs a second level check to determine the status of the Essbase server instance. The second level check try to login in the Essbase server instance to ensures that the processes are truly available for Essbase server instance.
- Depending upon the `MonitorProgram` attribute, the monitor function can perform a customized check using a user-supplied monitoring utility. See [“Executing a customized monitoring program”](#) on page 31.

Clean

In case of a failure or after an unsuccessful attempt to online or offline, the clean function removes any Essbase server processes remaining in the system.

The function performs the following tasks:

- Attempts to gracefully shut down the Essbase server instance.

- If a graceful shutdown fails, the clean function looks for all the processes running for the Essbase server instance, and cleans the processes by killing them.

Installing and configuring Essbase Server for high availability

This chapter includes the following topics:

- [About Essbase Server](#)
- [Uniquely identifying Essbase Server server instances](#)
- [About installing Essbase Server for high availability](#)
- [Configuring the Essbase Server for high availability](#)

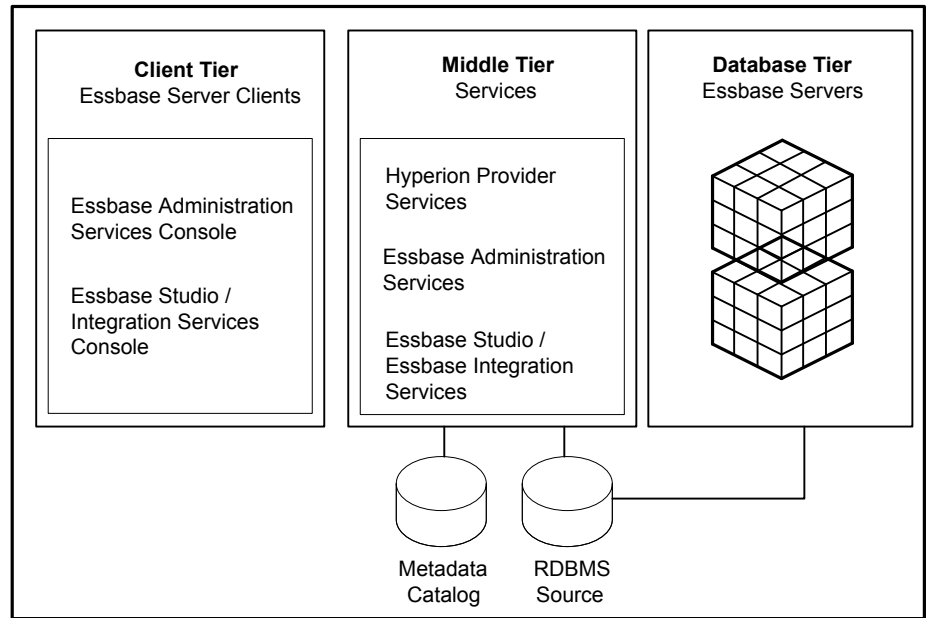
About Essbase Server

Oracle Essbase is a multi-threaded Online Analytical Processing (OLAP) database software that utilizes symmetric multiprocessing hardware platforms. The hardware platforms are based on Web-deployable, thin-client architecture. The server acts as a shared resource, handling all data storage, caching, calculations, and data security. The Essbase Server client needs to retrieve and view data that resides on a server. All Essbase application components, like database outlines, calculation scripts, application control, and multidimensional database information, reside on a server. Essbase allows you to configure server disk storage to span multiple disk drivers enabling you to store large databases.

Essbase server runs multi-threaded operating system to efficiently manage simultaneous requests and server agent process that act as a traffic coordinator for all the user requests that are sent to the applications. Aggregate storage databases provide an alternative to block storage. MaxL is a multidimensional

database access language that automates Essbase administration and maintenance tasks.

Figure 2-1 Oracle Essbase Architecture



Administration Services

It is a database and an interface to Essbase that provides a single point of access console to multiple Essbase Servers. It allows you to design, develop, maintain, and manage multiple Essbase servers, applications, and databases. It allows you to preview the data from within the console.

Essbase Studio

It simplifies the cube construction since it offers single environment to perform tasks related to data modeling, cube designing, and analytic application construction. It supports modeling of various data source types, which help in building Essbase applications. A single common metadata repository or catalog captures all the metadata related to all the Essbase applications built in the enterprise and allows the reuse of metadata at the lowest level of granularity.

Integration Services

It is an optional component that provides a metadata driven environment to bridge the gap between the data stored in Essbase databases and detailed data stored in relational databases. The Hybrid Analysis feature provides business users detail for decision-making and the IT managers' modularity in designing and maintaining large-scale analytic applications. The Hybrid Analysis stores parts of Essbase databases in a relational database. This data is mapped to appropriate Essbase hierarchies.

Provider Services

It provides a middle-tier data-source to Essbase for Java API, Smart View, and XMLA clients. It supports concurrent analytical scenarios and provides scalability and reliability in a distributed Web-enabled enterprise environment.

Uniquely identifying Essbase Server server instances

For multiple Essbase server instance running concurrently on a single node, the Veritas agent must be able to uniquely identify each of the Essbase instance on that system. Each Essbase server has a unique `ArborPath` location. The Veritas agent uses the `ArborPath` attribute value to identify the Essbase server instance uniquely.

Differentiating the Essbase server instances is important to identify each Essbase server uniquely. When the Veritas agent kills the processes of a non-responsive or failed Essbase instance in absence of unique `ArborPath` for each Essbase Server instance, the Veritas agent may kill processes for more than one Essbase server instance during a clean operation.

About installing Essbase Server for high availability

Install Essbase server on shared storage if you want the same installation to be shared among all the nodes in the cluster.

When installing Essbase Server, ensure that the user name, UNIX uid, group name, and UNIX gid for the Essbase installation user is the same on all the nodes.

The user and the group must be local and not Network Information Service (NIS and NIS+) users.

For more details, refer to the product documentation.

Configuring the Essbase Server for high availability

This section provides the information about the tasks you must perform to configure Essbase Server for high availability.

Synchronizing accounts and services

Ensure that you synchronize accounts and services in the following ways:

- Synchronize the Essbase installation user accounts user name, UNIX uid, group name, and UNIX gid across all nodes in the cluster.
- The `/etc/services` entries should be consistent on all cluster nodes.

Removing physical host dependencies

Perform the following tasks to remove the physical host dependencies:

- Add or modify the parameter `NODENAME` in the `essbase.cfg` file at the `bin` directory of Essbase instance `ArborPath` (`ArborPath/bin/essbase.cfg`).
`NODENAME essbase.vxindia.veritas.com`
- Modify the parameter in the `Server1in server.properties` file at the `bin` directory of Essbase instance `ArborPath` (`ArborPath/bin/server.properties`).
`Server1= essbase.vxindia.veritas.com`

Note: `essbase.vxindia.veritas.com` is the virtual host name of the Essbase server. It should be resolved from the network.

Installing, upgrading, and removing the agent for Essbase Server

This chapter includes the following topics:

- [Before you install the Veritas agent for Essbase Server](#)
- [About the ACC library](#)
- [Installing the ACC library](#)
- [Installing the agent in a VCS environment](#)
- [Removing the agent in a VCS environment](#)
- [Removing the ACC library](#)

Before you install the Veritas agent for Essbase Server

For VCS, do the following:

- Install and configure Veritas Cluster Server.
For more information on installing and configuring Veritas Cluster Server, refer to the *Veritas Cluster Server Installation Guide*.
- Install the latest version of ACC Library.
To install or update the ACC Library package, locate the library and related documentation in the agentpack tarball.
See [“Installing the ACC library”](#) on page 20.

About the ACC library

The operations of a VCS 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 agent's tar file has already been extracted.

Installing the ACC library

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

To install the ACC library

- 1 Log in as superuser.
- 2 Download ACC Library.

You can download either the complete Agent Pack tarball or the individual ACCLib tarball from the Symantec Operations Readiness Tools (SORT) site (<https://sort.symantec.com/agents>).

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

AIX	<code>cd1/aix/vcs/application/acc_library/version_library/pkgs</code>
HP-UX	<code>cd1/hpux/generic/vcs/application/acc_library/version_library/pkgs</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/pkgs</code>

where *dist_arch* is sol_sparc

- 4 If you downloaded the individual ACCLib tarball, navigate to the pkgs directory (for AIX, HP-UX, 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          # installp -ac -d VRTSacclib.bff VRTSacclib

HP-UX        # swinstall -s `pwd` VRTSacclib

Linux        # rpm -i \
              VRTSacclib-VersionNumber-GA_GENERIC.noarch.rpm

Solaris      # pkgadd -d VRTSacclib.pkg
```

Installing the agent in a VCS environment

Install the agent for Essbase Server on each node in the cluster.

To install the agent in a VCS environment

- 1 Download the agent from the Symantec Operations Readiness Tools (SORT) site: <https://sort.symantec.com/agents>.

You can download either the complete Agent Pack tarball or an individual agent tarball.

- 2 Uncompress the file to a temporary location, say /tmp.
- 3 If you downloaded the complete Agent Pack tarball, navigate to the directory containing the package for the platform running in your environment.

```
AIX          cd /aix/vcs/application/essbase_agent/
              vcs_version/version_agent/pkgs

HP-UX        cd /hpuX/generic/vcs/application/essbase_agent/
              vcs_version/version_agent/pkgs

Linux        cd /linux/generic/vcs/application/essbase_agent/
              vcs_version/version_agent/rpms

Solaris      cd /solaris/dist_arch/vcs/application/essbase_agent/
              vcs_version/version_agent/pkgs

              where, dist_arch is sol_sparc
```

If you downloaded the individual agent tarball, navigate to the pkgs directory (for AIX, HP-UX, and Solaris), or rpms directory (for Linux).

4 Log in as superuser.

5 Install the package.

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

```
HP-UX    # swinstall -s `pwd` VRTSvcsess
```

```
Linux    # rpm -ihv \  
          VRTSvcsess-AgentVersion-GA_GENERIC.noarch.rpm
```

```
Solaris  # pkgadd -d . VRTSvcsess
```

6 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 25.

Removing the agent in a VCS environment

You must uninstall the agent for Essbase Server from a cluster while the cluster is active.

To uninstall the agent in a VCS environment

1 Log in as a superuser.

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

```
# haconf -makerw
```

3 Remove all Essbase resources from the cluster. Use the following command to verify that all resources have been removed:

```
# hares -list Type=Essbase
```

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

```
# hatype -delete Essbase
```

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

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

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

- 6 Use the platform's native software management program to remove the agent for Essbase Server from each node in the cluster.

Execute the following command to uninstall the agent:

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

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 VRTSaccclib
HP-UX	# swremove VRTSaccclib
Linux	# rpm -e VRTSaccclib
Solaris	# pkgrm VRTSaccclib

Configuring the agent for Essbase Server

This chapter includes the following topics:

- [About configuring the Veritas agent for Essbase Server](#)
- [Importing the agent types files in a VCS environment](#)
- [Essbase server agent attributes](#)
- [Executing a customized monitoring program](#)

About configuring the Veritas agent for Essbase Server

After installing the Veritas agent for Essbase Server, you must import the agent type configuration file. After importing this file, you can create and configure an Essbase resource. Before you configure a resource, review the attributes table that describes the resource type and its attributes.

See [“About sample configurations for the agent for Essbase Server”](#) on page 43.

Importing the agent types files in a VCS environment

To use the agent for Essbase Server, you must import the agent types file into the cluster.

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

- 1** Start the Veritas Cluster Manager and connect to the cluster on which the agent is installed.
- 2** Click **File > Import Types**.
- 3** In the Import Types dialog box, select the following file:

VCS 4.x	■ AIX	/etc/VRTSvcS/conf/sample_Essbase/
	■ HP-UX	EssbaseTypes.cf
	■ Linux	
	■ Solaris	
VCS 5.x	■ AIX	/etc/VRTSagents/ha/conf/Essbase/
	■ HP-UX	EssbaseTypes.cf
	■ Linux	
VCS 5.0	■ Solaris SPARC	/etc/VRTSagents/ha/conf/Essbase/ EssbaseTypes50.cf
VCS 5.1	■ Solaris SPARC	/etc/VRTSagents/ha/conf/Essbase/ EssbaseTypes51.cf

- 4** Click **Import**.
- 5** Save the VCS configuration.

The Essbase agent type is now imported to the VCS engine.

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

To import the agent types file using the Veritas Cluster Server command line interface (CLI), perform the following steps.

- 1** Log on to any one of the systems in the cluster as the superuser.
- 2** Create a temporary directory.


```
# mkdir ./temp
```

```
# cd ./temp
```
- 3** Copy the sample file Types.cf.

VCS 4.x	■ AIX	/etc/VRTSvcs/conf/sample_Essbase/
	■ HP-UX	EssbaseTypes.cf
	■ Linux	
	■ Solaris	
VCS 5.x	■ AIX	/etc/VRTSagents/ha/conf/Essbase/
	■ HP-UX	EssbaseTypes.cf
	■ Linux	
VCS 5.0	■ Solaris SPARC	/etc/VRTSagents/ha/conf/Essbase/ EssbaseTypes50.cf

The following example assumes VCS 5.0 is installed on Solaris.

```
# cp /etc/VRTSagents/ha/conf/Essbase/EssbaseTypes.cf .
```

4 Create a dummy main.cf file:

```
# echo 'include "EssbaseTypes.cf"' > main.cf
```

5 Create the Essbase resource type as follows:

```
# hacf -verify .  
  
# haconf -makerw  
  
# sh main.cmd  
  
# haconf -dump
```

The Essbase agent type is now imported to the VCS engine.

You can now create Essbase resources. For additional information about using the VCS CLI, refer to the *Veritas Cluster Server User's Guide*.

Essbase server agent attributes

Refer to the required attributes and optional attributes while configuring the agent for Essbase server.

[Table 4-1](#) lists the required attributes for the Essbase server.

Table 4-1 Required attributes

Required attribute	Description
ArborPath	<p>Specifies the base directory for Essbase server installation. This directory contains the binaries and configuration directories relative to ARBORPATH.</p> <p>Type and Dimension: string-scaler</p> <p>Default: ""</p> <p>Example: /oracle/hyperion/AnalyticServices</p>
HostName	<p>Specifies the virtual host name for Essbase server instance. ESSCMD uses the host name to log on to the Essbase Server.</p> <p>Type and Dimension: string-scaler</p> <p>Default: ""</p> <p>Example: web1.symantec.com</p>
User	<p>Specifies the user name that the Veritas agent uses for executing the programs for managing a Essbase Server. If unspecified, the server instance starts as user root.</p> <p>The user name must be synchronized across the systems in the cluster. The user name must resolve to the same UID and must have the same default shell on each system in the cluster. The Veritas agent entry points use the getpwnam(3c) function call to obtain UNIX user attributes. As a result, the user can be defined locally or can be defined in a common repository (that is, NIS, NIS+, or LDAP). In the latter case, the agent fails if the access to this repository fails.</p> <p>The supported shell environments are: ksh, sh, and csh.</p> <p>Type and Dimension: string-scaler</p> <p>Default: ""</p> <p>Example: cvradm</p>
EssbaseUser	<p>Specifies the Essbase user which is configured for Essbase server instance. The Veritas agent uses this user to log on to the Essbase server using the ESSCMD command.</p> <p>Type and Dimension: string-scaler</p> <p>Default: ""</p> <p>Example: essuser</p>

Table 4-1 Required attributes (*continued*)

Required attribute	Description
EssbasePassword	<p>Specifies the encrypted Essbase user password to start the Essbase server and log on to the Essbase server using the <code>ESSCMD</code> command.</p> <p>Type and Dimension: string-scaler</p> <p>Default: ""</p> <p>Example: password</p>
EnvFile	<p>Specifies the full path to the file that must be sourced with the UNIX shell. Source this file to set the environment before executing any Essbase server command.</p> <p>Symantec recommends storing this file on a shared disk.</p> <p>The supported shell environments are: ksh, sh, and csh.</p> <p>Type and Dimension: string-scaler</p> <p>Default: ""</p> <p>Example: /Essbase/products/Essbase/EssbaseServer/hyperionenv.env</p>
ResLogLevel	<p>Specifies the logging detail performed by the agent for the resource.</p> <p>The valid values are as follows:</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-scaler</p> <p>Default: INFO</p> <p>Example: TRACE</p>
ComponentType	<p>Specifies the name of the component for specific component types.</p> <p>Type and Dimension: string-scaler</p> <p>Default: EssbaseServer</p> <p>Example: EssbaseServer</p>

[Table 4-2](#) lists the optional attributes for the Essbase server.

Table 4-2 Optional attributes

Optional attribute	Description
SecondLevelMonitor	<p>Used to enable second-level monitoring. Second-level monitoring is a deeper, more thorough state check of the Essbase server. 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. This interpretation may be extended to other values.</p> <p>Note: Exercise caution while setting SecondLevelMonitor to large numbers. For example, if the MonitorInterval is set to 60 seconds and the SecondLevelMonitor is set to 100, then the second level check is executed every 100 minutes, which may not be as often as intended. For maximum flexibility, no upper limit is defined for SecondLevelMonitor.</p> <p>Type and Dimension: integer-scaler</p> <p>Default: 0</p> <p>Example: 5</p>
MonitorProgram	<p>Specifies the absolute path name of an external, user-supplied monitor executable.</p> <p>See “Executing a customized monitoring program” on page 31.</p> <p>Type and Dimension: string-scaler</p> <p>Default: ""</p> <p>Example: /Essbase/products/Essbase/EssbaseServer/myMonitor.sh</p>
Port	<p>Specifies the port number on which this Essbase server instance is listening. To determine whether the HTTP server is responding to Essbase requests, the monitor entry point uses the value specified by the Port attribute.</p> <p>When second level monitor is enabled, this attribute is specified.</p> <p>Type and Dimension: integer-scaler</p> <p>Default: 80</p> <p>Example: 8080</p>

Executing a customized monitoring program

The monitor function executes a custom monitor program to perform an additional Essbase Server state check. The monitor function executes the utility specified in the MonitorProgram attribute, if:

- The specified utility is a valid executable file.
- The first level process check indicates that the Essbase Server instance is online.
- The SecondLevelMonitor attribute is either set to 0 or 1, and the second level check indicates that the Essbase Server instance is online.
- The SecondLevelMonitor attribute is set to greater than 1, but the second level check is deferred for this monitoring cycle.

The monitor function interprets the utility exit code as follows:

110 or 0	Essbase server instance is online
100 or 1	Essbase server instance is offline
99	Essbase server instance is unknown
Any other value	Essbase server instance is unknown

To ensure that the customized utility is always available to the agent, Symantec recommends storing the file in a shared directory that is available on an online node.

Configuring the service groups for Essbase Server

This chapter includes the following topics:

- [About configuring service groups for Essbase Server](#)
- [Before configuring the service groups for Essbase Server](#)
- [Configuring service groups for Essbase Server](#)
- [Generating the environments file for Essbase Server](#)

About configuring service groups for Essbase Server

Configuring the Essbase Server service group involves creating the Essbase 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 Operations Manager
- The command-line

See [“Configuring service groups for Essbase Server”](#) on page 34.

Before configuring the service groups for Essbase Server

Before you configure the Essbase service group, you must:

- Verify that VCS is installed and configured on all nodes in the cluster where you will configure the service group.
Refer to the *Veritas Cluster Server Installation Guide* for more information.
- Verify that Essbase server is installed and configured on all the systems in the cluster on the same path that will host a Essbase service group.
See [“About installing Essbase Server for high availability”](#) on page 17.
- Verify that the Veritas agent for Essbase Server is installed on all nodes in the cluster.
See [“Installing the agent in a VCS environment”](#) on page 21.
- Verify that the type definition for the Veritas agent for Essbase Server is imported into the VCS engine.
See [“Importing the agent types files in a VCS environment”](#) on page 25.

Configuring service groups for Essbase Server

While setting up a cluster, you must ensure that the cluster has some spare capacity to handle the Essbase Server failover scenarios.

The cluster should be able to provide application failover by encapsulating the resources required for an application into a service group. A service group is a virtualized application that can switch between the cluster nodes. It contains a set of dependent resources, such as IP addresses, NIC cards, and dependent application processes. It also includes logic about the dependencies between the application components.

These service groups should thus be configured such that the cluster can start, stop, monitor, and switch the service groups between the nodes, depending upon the server faults or resource faults. An administrator should also be proactively able to move a service group between cluster nodes to perform preventative maintenance or apply patches.

Perform the following steps to add a service group for Essbase Server

- 1 Create a service group for Essbase server.

```
# hagrps -add Essbase_grp
```

For more details on creating a service group refer to, *Veritas Cluster Server User's Guide*.

- 2 Modify the SystemList attribute for the group, to add systems.

For example,

```
# hagrps -modify Essbase_grp SystemList systemA 0 systemB 1
```

3 Create resources for NIC and IP.

For example,

```
# hares -add Essbase_nic NIC Essbase_grp
# hares - Essbase_ip IP Essbase_grp
```

For more details on creating and modifying resource attributes for NIC and IP, refer to, *Bundled Agents Reference Guide*.

4 Create links between the resources.

For example,

```
# hares -link Essbase_ip Essbase_nic
```

5 If you have not installed the Essbase server on shared file system then follow the steps 6 onwards, otherwise, directly go to step 9 and continue further.

6 Create a separate file system for Essbase Server on shared disk.

7 Copy the contents of installation directory of the Essbase to the shared file system.

8 Delete the contents of installation directory and then create a link between installation directory and the directory on the shared disk where you have copied the contents of the installation directory.

9 Add the file system to respective agent service group using Mount, DiskGroup, and Volume resources.

10 Create Mount and DiskGroup resources.

For example,

```
# hares -add Essbase_dg DiskGroup Essbase_grp
# hares - Essbase_mount Mount Essbase_grp
```

Based on the Essbase server instance you are clustering, modify the resource attributes of Mount and DiskGroup resources.

For more details on creating and modifying resource attributes for Mount and DiskGroup refer to, *Bundled Agents Reference Guide*.

11 Create links between the resources.

For example,

```
# hares -link Essbase_mount Essbase_dg
```

12 Create Essbase resource for Essbase server.

For example,

```
# hares -add Essbase_res Essbase Essbase_grp
```

Based on the Essbase server instance you are clustering, modify the resource attributes. For more information on agent attributes,

See [“Essbase server agent attributes”](#) on page 27.

13 Create resource dependencies for Essbase resource.

The Essbase resource depends on the IP and Mount resources.

```
# hares -link Essbase_res Essbase_ip
```

```
# hares -link Essbase_res Essbase_mount
```

14 Verify the final resource dependencies for Essbase_grp server group.

For example,

```
# hares -dep
```

Group	Parent	Child
Essbase_grp	Essbase_res	Essbase_ip
Essbase_grp	Essbase_res	Essbase_mount
Essbase_grp	Essbase_ip	Essbase_nic
Essbase_grp	Essbase_mount	Essbase_dg

Generating the environments file for Essbase Server

To generate the environments file for Essbase Server

1 Login as Essbase installation user using the following command.

```
su - essadmin
```

2 Capture the environment with the following command.

```
env > /home/essadmin/essadmin.env
```

3 Modify the file according to the Essbase installation user shell environment.

For example, if the generated file contains environments for bash shell and Essbase installation user shell is C shell, convert the file to C shell environments.

- Edit the essdmin.env file to add string 'setenv' at the beginning of each line.

- Replace the '=' with space " " in the file.
- 4 Copy the essadmin.env file to shared directory and use it as the Essbase Server instance environments file in EnvFile attribute. Ensure that the permissions are set properly for Essbase user.

```
chmod 755 essadmin.env
```

Note: Before generating the EnvFile, verify the successful execution of start, stop, and second level monitor command with Essbase user environment.

Troubleshooting the agent for Essbase Server

This chapter includes the following topics:

- [Using the correct software and operating system versions](#)
- [Meeting prerequisites](#)
- [Configuring Essbase resources](#)
- [Starting the Essbase instance outside a cluster](#)
- [Reviewing error log files](#)

Using the correct software and operating system versions

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

See [“Supported software”](#) on page 10.

Meeting prerequisites

Before installing the agent for Essbase Server, double check that you meet the prerequisites.

For example, you must install the ACC library on VCS before installing the agent for Essbase Server.

See [“Before you install the Veritas agent for Essbase Server”](#) on page 19.

Configuring Essbase resources

Before using an Essbase agent resource, ensure that you configure the resource properly. For a list of attributes used to configure all Essbase resources, refer to the agent attributes.

See [“Essbase server agent attributes”](#) on page 27.

Starting the Essbase 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 Essbase instance independent of the cluster framework. Refer to the cluster documentation for information about disabling a resource.

You can then restart the Essbase 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 an Essbase server instance outside the cluster framework, is illustrated as follows.

To restart the Essbase server outside the framework

- 1 Log in to the Essbase node as an User.

```
# su - User
```

- 2 Source the environment file.

```
# . EnvFile
```

- 3 Start the Essbase server.

```
# ArborPath/bin/ESSBASE -b secure <<EOF &
```

```
EssbasePassword
```

```
EOF
```

If the Essbase server works properly outside the cluster framework, you can then attempt to implement the Essbase server within the cluster framework.

Reviewing error log files

If you face problems while using Essbase or the agent for Essbase Server, use the log files described in this section to investigate the problems.

Reviewing cluster log files

In case of problems while using the agent for Essbase Server, 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`.

You can also access the Essbase agent log file for more detailed information. The agent log file is located at `/var/VRTSvcS/log/Essbase_A.log`

Using trace level logging

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

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

To localize `ResLogLevel` attribute for a resource

- 1 Identify the resource for which you want to enable detailed logging.
- 2 Localize the `ResLogLevel` attribute for the identified resource:

```
# hares -local Resource_Name ResLogLevel
```

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

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

- 4 Note the time before you begin to operate the identified resource.
- 5 Test the identified resource. The function reproduces the problem that you are attempting to diagnose.
- 6 Note the time when the problem is reproduced.

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

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

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

You can also contact Symantec support for more help.

Sample Configurations

This appendix includes the following topics:

- [About sample configurations for the agent for Essbase Server](#)
- [Sample agent type definition](#)
- [Sample configuration](#)
- [Sample configuration with Solaris zone support](#)
- [Sample service group configuration](#)
- [Sample service group configuration with Solaris zone support](#)

About sample configurations for the agent for Essbase Server

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

Sample agent type definition

This section lists the sample agent type definition files for ESSBase agent on different versions of VCS.

After importing the agent types into the cluster, save the configuration file on the system disk using the `haconf -dump` command. The `EssbaseTypes.cf` file is found in the `/etc/VRTSvcs/conf/config` cluster configuration directory.

For VCS 4.x

```
type Essbase (
static str ArgList[] = { ResLogLevel, State, IState, User, HostName,
Port, EnvFile, ArborPath, ComponentType, EssbaseUser, EssbasePassword,
SecondLevelMonitor, MonitorProgram }
    str ResLogLevel = INFO
    str User = "root"
    str HostName
    int Port = 1423
    str EnvFile
    str ArborPath
    str ComponentType = "EssbaseServer"
    str EssbaseUser
    str EssbasePassword
    int SecondLevelMonitor = 0
    str MonitorProgram
)
```

For 5.0

```
type Essbase (
    static str ContainerType = Zone
    static boolean AEPTIMEOUT = 1
    static str AgentFile = "/opt/VRTSvcs/bin/Script50Agent"
    static str AgentDirectory = "/opt/VRTSagents/ha/bin/Essbase"
static str ArgList[] = { ResLogLevel, State, IState, User, HostName,
Port, EnvFile, ArborPath, ComponentType, EssbaseUser, EssbasePassword,
SecondLevelMonitor, MonitorProgram }
    str ResLogLevel = INFO
    str User = "root"
    str HostName
    int Port = 1423
    str EnvFile
    str ArborPath
    str ComponentType = "EssbaseServer"
    str EssbaseUser
    str EssbasePassword
    int SecondLevelMonitor = 0
    str MonitorProgram
    str ContainerName
)
```

For 5.1

```

type Essbase (
    static int ContainerOpts{} = { RunInContainer=1, PassCInfo=0 }
    static boolean AEPTIMEOUT = 1
    static str AgentFile = "/opt/VRTSvcs/bin/Script51Agent"
    static str AgentDirectory = "/opt/VRTSagents/ha/bin/Essbase"
    static str ArgList[] = { ResLogLevel, State, IState, User, HostName,
    Port, EnvFile, ArborPath, ComponentType, EssbaseUser, EssbasePassword,
    SecondLevelMonitor, MonitorProgram }
    str ResLogLevel = INFO
    str User = "root"
    str HostName
    int Port = 1423
    str EnvFile
    str ArborPath
    str ComponentType = "EssbaseServer"
    str EssbaseUser
    str EssbasePassword
    int SecondLevelMonitor = 0
    str MonitorProgram
)

```

Sample configuration

This section provides a sample configuration for Essbase server agent. The sample configuration depicts a graphical view of the resource types, resources, and resource dependencies within the service group.

```

include "types.cf"
include "EssbaseTypes51.cf"

cluster system01_02 (
    UserNames = { admin = IhiAhcHeiDiiGqiChf }
    Administrators = { admin }
    HacliUserLevel = COMMANDROOT
)

system system01 (
)

system system02 (
)

group Essbase_grp (
    SystemList = { system01 = 0, system02 = 1 }
)

```

```

)

DiskGroup Essbase_dg (
    DiskGroup = Essbase_dskgrp
)

Essbase Essbase_res (
    HostName = "essbase.vxindia.veritas.com"
    EnvFile = "/Essbase/products/Essbase/
                EssbaseServer/hyperionenv.env"
    ArborPath = "/Essbase/products/Essbase/EssbaseServer"
    EssbaseUser = isv
    EssbasePassword = cqiOfqFqiOfqF
    SecondLevelMonitor = 1
    MonitorProgram = "/Essbase/monitor.sh"
)

IP Essbase_ip (
    Device = bge0
    Address = "100.109.173.10"
    NetMask = "255.255.254.0"
)

Mount Essbase_mount (
    MountPoint = "/Essbase"
    BlockDevice = "/dev/vx/dsk/Essbase_dskgrp/Essbase_vol"
    FSType = vxfs
    FsckOpt = "-y"
)

NIC Essbase_nic (
    Device = bge0
)

Essbase_ip requires Essbase_nic
Essbase_mount requires Essbase_dg
Essbase_res requires Essbase_ip
Essbase_res requires Essbase_mount

// resource dependency tree
//
//     group Essbase_grp
//     {

```

```

//      Essbase Essbase_res
//      {
//      IP Essbase_ip
//      {
//      NIC Essbase_nic
//      }
//      Mount Essbase_mount
//      {
//      DiskGroup Essbase_dg
//      }
//      }
//      }

```

Sample configuration with Solaris zone support

This section provides a sample configuration with Solaris local zone support for Oracle Essbase Server agent.

```

include "types.cf"
include "EssbaseTypes51.cf"

cluster hypeclus (
    UserNames = { admin = HMNfMHmJNiNNlVNhMK }
    Administrators = { admin }
)

system vcssx082 (
)

system vcssx084 (
)

group ESSBASE-SG (
    SystemList = { vcssx082 = 0, vcssx084 = 1 }
)

DiskGroup ESS-DG (
    Critical = 0
    DiskGroup = infomixdb
)

Essbase ESSBASE-RES (
    Critical = 0

```

```
        ResLogLevel = TRACE
        User = hyperion
        HostName = essbase
        Port = 13450
        EnvFile = "/export/home/hyperion/.cshrc"
        ArborPath = "/export/home/hyperion/products/Essbase/EssbaseServ
EssbaseUser = hyperion
        EssbasePassword = ftlRitIlglhll
    )

IP ESS-IP (
    Critical = 0
    Device = bge0
    Address = "10.209.62.60"
)

Mount ESS-MNT (
    Critical = 0
    MountPoint = "/export/home/hyperion"
    BlockDevice = "/dev/vx/dsk/infomixdb/infomixdb_vol"
    FSType = vxfs
    FsckOpt = "-y"
)

NIC ESS-NIC (
    Critical = 0
    Device = bge0
)

Volume ESS-VOL (
    Critical = 0
    Volume = infomixdb_vol
    DiskGroup = infomixdb
)

Zone ESSBASE-ZONE (
    Critical = 0
)

ESS-IP requires ESSBASE-ZONE
ESS-MNT requires ESS-VOL
ESS-VOL requires ESS-DG
ESSBASE-RES requires ESS-IP
```



```
ESSBASE-ZONE requires ESS-MNT
ESSBASE-ZONE requires ESS-NIC

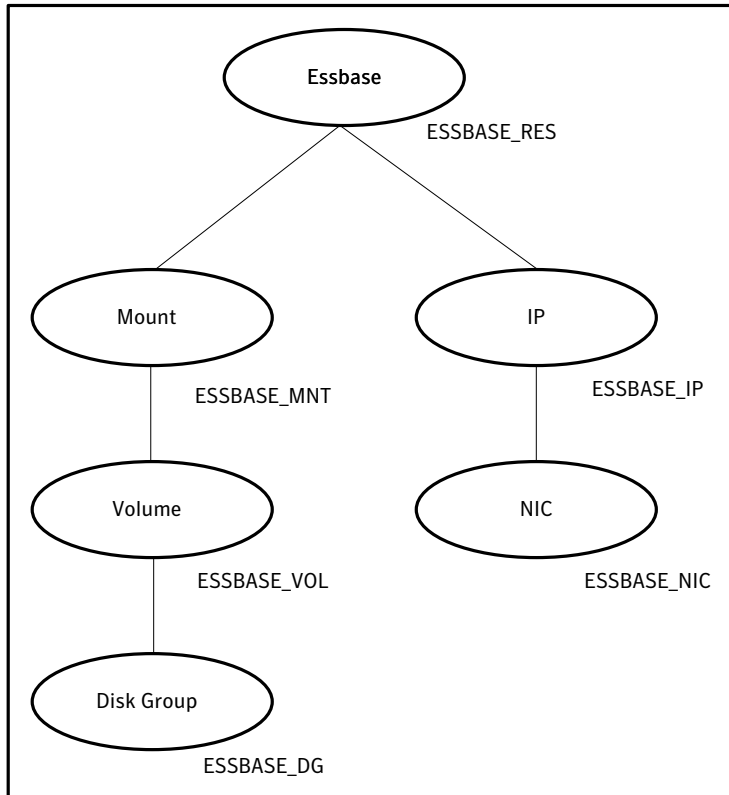
// resource dependency tree
//
//      group ESSBASE-SG
//      {
//      Essbase ESSBASE-RES
//      {
//      IP ESS-IP
//      {
//      Zone ESSBASE-ZONE
//      {
//      Mount ESS-MNT
//      {
//      Volume ESS-VOL
//      {
//      DiskGroup ESS-DG
//      }
//      }
//      NIC ESS-NIC
//      }
//      }
//      }
//      }
```

Sample service group configuration

This section includes sample service groups configurations in a VCS environment.

The service group includes a DiskGroup resource, a NIC resource, an IP resource, a Volume resource, and Mount resource, along with an Essbase Server resource.

Figure A-1 Sample service group for an Essbase Server instance



For Essbase Type definition See [“Sample agent type definition”](#) on page 43.

Sample service group configuration with Solaris zone support

This section includes sample service groups with Solaris zone support.

The figure shows a service group with Oracle Essbase Server instance installed, configured, and running under Solaris local zones.

Figure A-2 Sample service group with Oracle Essbase Server instance

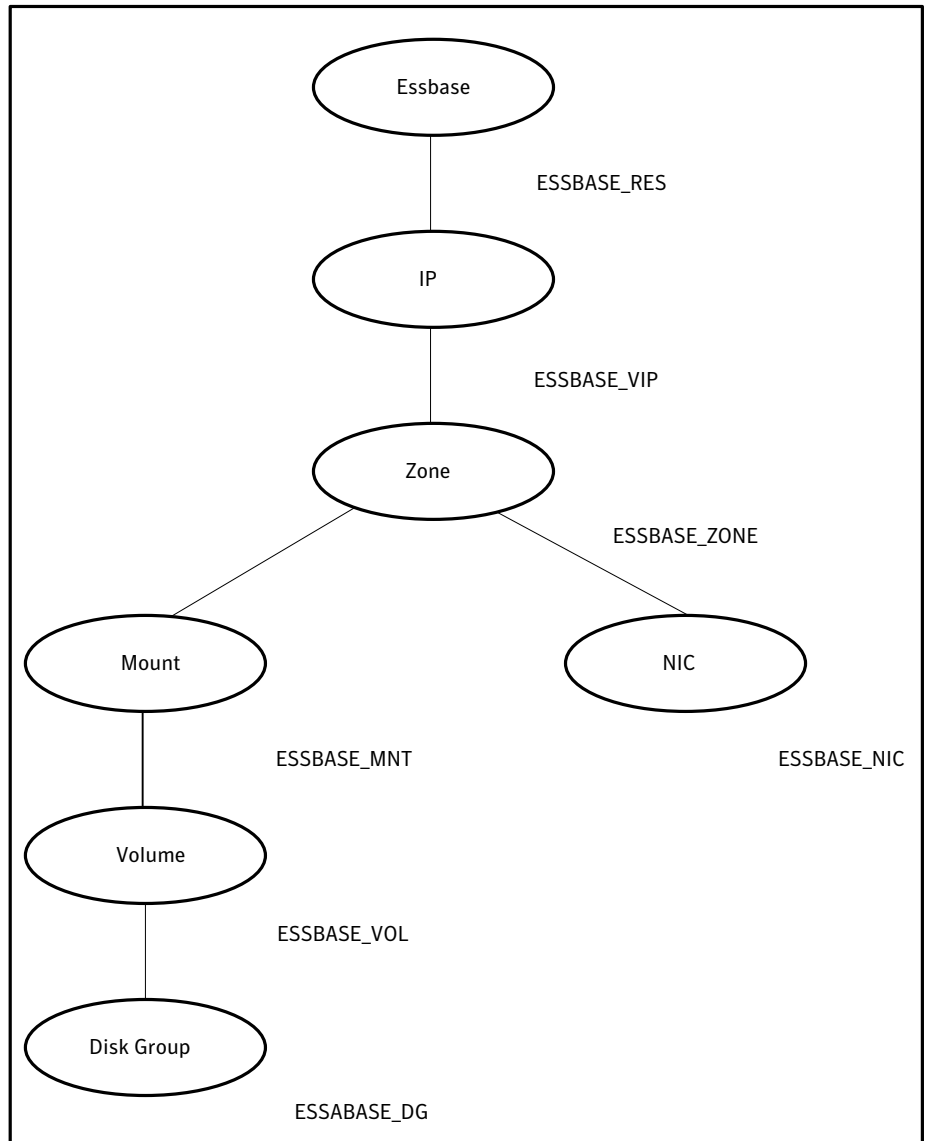
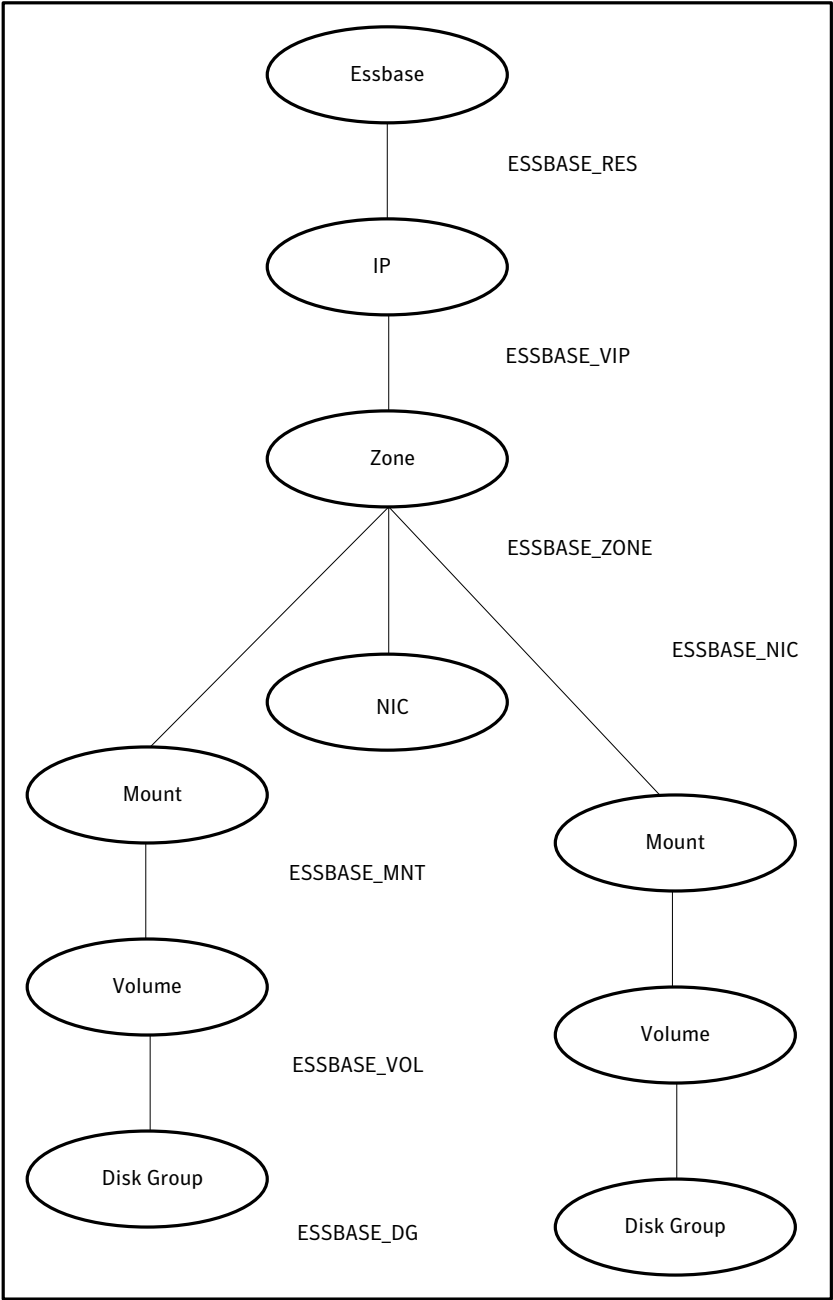


Figure A-3

Sample service group with Oracle Essbase Server instance installed in a shared disk and local zone configured on a shared disk



Index

A

- about
 - configuring service groups 33
 - Essbase server 15
 - installing Essbase Server for high availability 17
- about ACC library 20
- ACC library
 - installing 20
 - removing 23
- agent
 - configuring service groups 34
 - importing agent types files 25
 - installing, VCS environment 21
 - optional attributes 29
 - overview 9
 - required attributes 27
 - supported software 10
 - uninstalling, VCS environment 22
 - what's new 10
- agent attributes 27
 - Arbopath 28
 - ComponentType 29
 - EnvFile 29
 - EssbasePassword 29
 - EssbaseUser 28
 - HostName 28
 - MonitorProgram 30
 - Port 30
 - ResLogLevel 29
 - SecondLevelMonitor 30
 - User 28
- agent configuration file
 - importing 25
- agent functions 11
- agent installation
 - general requirements 19
 - steps to install 21

B

- before
 - configuring the service groups 33

C

- Configuring
 - Essbase Server 18
- configuring monitor function 31

E

- environments file 36
- Essbase
 - configuring resources 40
 - starting instance outside cluster 40
- executing custom monitor program 31

L

- logs
 - reviewing cluster log files 41
 - reviewing error log files 41
 - using trace level logging 41

R

- removing agent, VCS environment 22

S

- sample agent type definition 43
- sample configuration files 45
 - Solaris zone support 47
- sample service group configuration 49
- service group
 - sample configurations, Solaris zone support 50
- starting the Essbase instance outside a cluster 40
- supported software 10

T

- troubleshooting
 - meeting prerequisites 39
 - reviewing error log files 41
 - reviewing cluster log files 41
 - using trace level logging 41
 - using correct software 39

U

uninstalling agent, VCS environment 22