Cluster Server Agent for SAP liveCache Installation and Configuration Guide

AIX, Linux, Solaris

8.0

VERITAS™

Last updated: 2022-04-08

Legal Notice

Veritas Technologies LLC
2625 Augustine Drive
Santa Clara, CA 95054
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:

infoscaledocs@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

Introducing the agent for SAP liveCache

This chapter includes the following topics:

- About the Cluster Server agent for SAP liveCache

- Supported software

- How the agent makes SAP liveCache highly available

- SAP liveCache agent functions

- Typical SAP liveCache configuration in a VCS cluster

- Setting up SAP liveCache in a VCS cluster

About the Cluster Server agent for SAP liveCache

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 SAP liveCache provides high availability for SAP liveCache Server in a cluster.

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 SAP liveCache supports, see the Veritas Services and Operations Readiness Tools (SORT) site: https://sort.veritas.com/agents.

How the agent makes SAP liveCache highly available

The Cluster Server agent for SAP liveCache continuously monitors the SAP liveCache server 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 SAP liveCache server 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 SAP liveCache server. The agent detects application failure if the monitoring routine reports an improper function of the SAP liveCache server processes. When this application failure occurs, the SAP liveCache server service group fails over to another node in the cluster.
  Thus, the agent ensures high availability for SAP liveCache servers.

SAP liveCache 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 a preliminary check to ensure that the SAP liveCache instance is not online on the specified node in the cluster.

- Forcibly removes any SAP liveCache processes that remain because of an unclean shutdown.

- Removes all the IPC resources for the SAP liveCache Instance based on the `/sapdb/data/ipc` (SAP standard) directory.

- Starts the SAP liveCache client x_server on the system.

- Starts the SAP liveCache instance using the `startrfc or sapevt` program.

- Ensures that the SAP liveCache instance is initialized successfully.

Offline

The offline function performs the following tasks:

- Performs a preliminary check to ensure that the SAP liveCache instance is not already offline on the specified node in the cluster.

- Stops the SAP liveCache instance using the `startrfc or sapevt` program.

- Removes all the IPC resources for the SAP liveCache instance based on the `/sapdb/data/ipc` (SAP standard) directory, if it exists.

- Ensures that the liveCache instance is stopped successfully.

Monitor

The monitor function monitors the state of the SAP liveCache instance on all nodes in the cluster. The function performs the following tasks:

- Scans the process table to verify the SAP liveCache instance processes are running.
  See "Monitoring SAP liveCache instance" on page 13.

- If the SecondLevelMonitor attribute is greater than 0, the monitor function performs a more thorough check of the SAP liveCache instance. The dbmcli utility is used to perform this second level check.

- Executes a custom monitor utility that the MonitorProgram attribute specifies.

Clean

The clean function performs the following tasks:

- Attempts to gracefully shutdown the SAP liveCache server with `dbmcli`.

- Waits for 10 seconds for the SAP liveCache instance to shut down successfully.

- Ensures that no relevant SAP liveCache server processes are running. If any processes remain, the function kills the remaining processes using a SIGKILL signal.

- Removes all IPC resources for the SAP liveCache instance based on the `/sapdb/data/ipc` (SAP Standard) directory.

Typical SAP liveCache configuration in a VCS cluster

A typical SAP liveCache configuration in a Cluster Server cluster has the following characteristics:

- VCS is installed and configured in a two-node cluster.
- The `/sapdb/LCName` directory is installed on shared storage.
- The SAP liveCache binaries are installed locally on both nodes or on shared disks.
- The Cluster Server agent for SAP liveCache is installed on both the nodes.

Figure 1-1 depicts a configuration where SAP liveCache instance binaries and /sapdb/LCName are installed completely on shared disks

Figure 1-1   SAP liveCache configuration
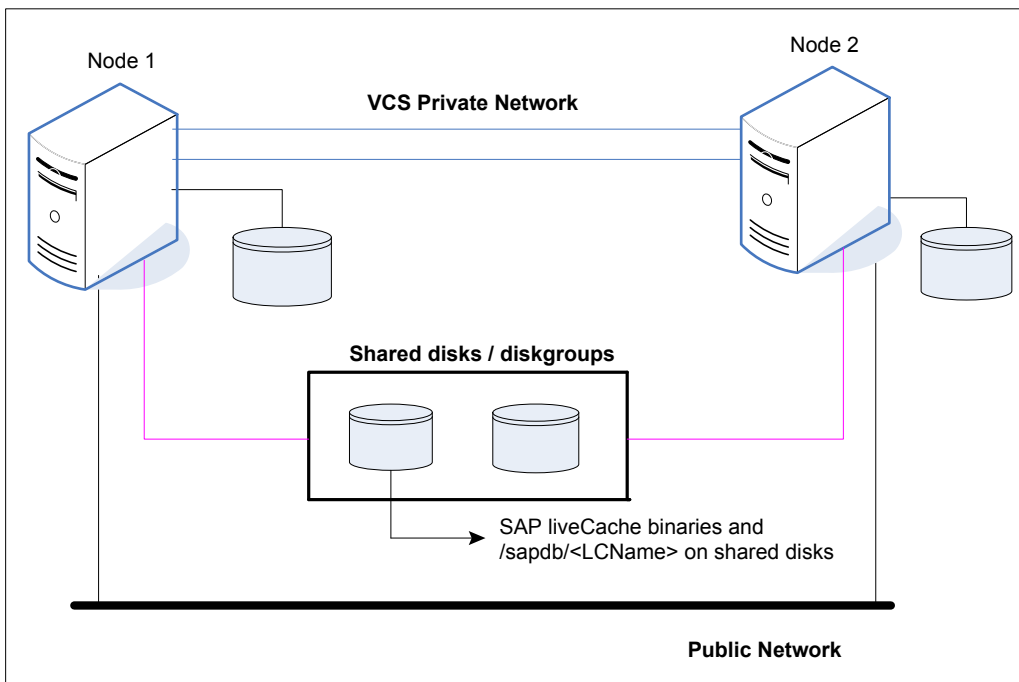
Figure 1-2   depicts a configuration where SAP liveCache instance binaries are installed locally on each node and /sapdb/LCName is installed on shared disks.
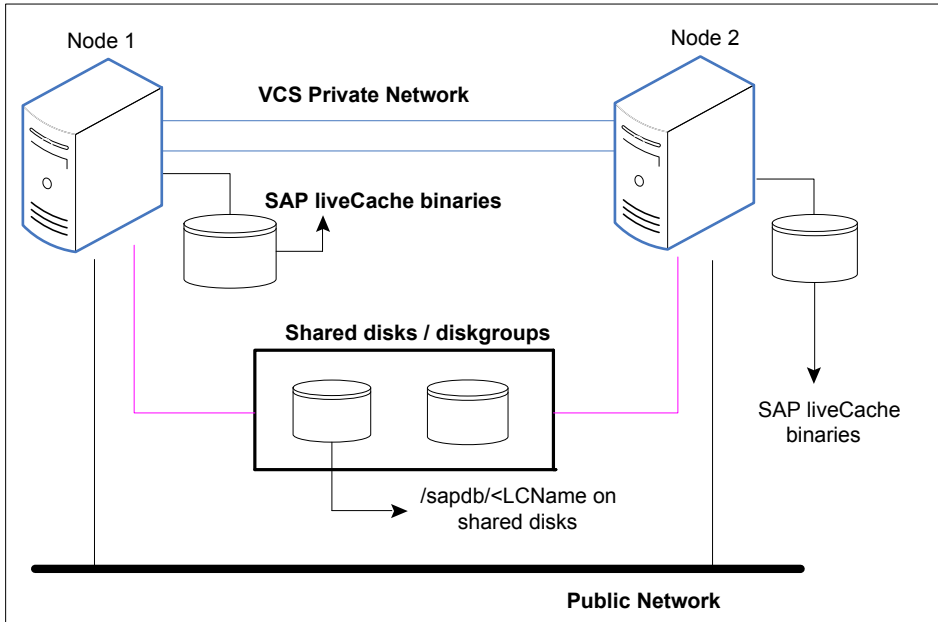
Setting up SAP liveCache in a VCS cluster

Perform the following tasks to set up SAP liveCache in a cluster:

1. Set up a VCS cluster.

2. Install and configure SAP liveCache for high availability.

3. Install the Cluster Server agent for SAP liveCache.

   See "Installing the agent in a VCS environment" on page 24.

4. Configure the service groups for SAP liveCache.

   See "About configuring service groups for SAP liveCache" on page 43.

Installing and configuring SAP liveCache for high availability

This chapter includes the following topics:

About SAP liveCache server

SAP liveCache is based on a memory-centric offshoot of the SAP DB technology that is shipped with SAP Advanced Planning and Optimization (APO) since Release 2.0. For the most resource-intensive planning questions, SAP APO pushes performance-critical application logic to SAP liveCache. The data that is required for those processes is also pushed to SAP liveCache, where it is kept persistent. The persistence of both data and application logic is beneficial as it allows different processes to work on the same data and avoids bottlenecks by following the paradigm "run the logic where the data is".

SAP liveCache - technology

SAP liveCache technology is an object-based enhancement of the MaxDB database system. MaxDB is a relational database system that was developed for Online Transaction Processing (OLTP). OLTP (MaxDB) and SAP liveCache database systems are variations of the same database system.

SAP liveCache was developed to manage complex objects, for example, in logistical solutions, such as SAP Supply Chain Management (SCM)/APO. In such solutions, large volumes of data must be permanently available and modifiable. You can use SAP liveCache technology to represent data structures and data flows, such as networks and relationships more easily and effectively. In SAP liveCache database, all data which must be accessible to the database system is located in the main memory. SAP liveCache database instances can only be used within SAP systems.

Uniquely identifying SAP liveCache server instances

You can virtualize a SAP liveCache instance using a cluster. Using shared disk and virtual IP addresses, you can manage a large set of SAP liveCache instances in a single cluster.

For multiple instances running concurrently on a single node, the agent must be able to uniquely identify each SAP liveCache instance on a single host system.

Each instance has a unique instance name.

The instance names may follow the conventional naming conventions as follows:

- Instance name should contain only three alphanumeric uppercase characters.

- Instance name should start with alphabet always.

- Instance name should be unique on the cluster.

Note that the LCName attribute forms a unique identifier that can identify the processes running for a particular SAP liveCache server instance.

Some examples are HOT, LC1, and so on.

Monitoring SAP liveCache instance

The monitor operation performs process level check to ensure the proper functioning of an SAP liveCache instance. The monitor operation checks for the processes from the process table which contains "*kernel LCName*" in their names.

It also checks the database state with "`dbmcli`" utility if *SecondLevelMonitor* attribute is set.

About installing SAP liveCache for high availability

You can install SAP liveCache in the following ways, in a VCS environment:

SAP liveCache on a shared disk	Install the SAP liveCache binaries and /sapdb/LCName on the shared disks.
SAP liveCache on a local disk	Install the SAP liveCache binaries on each node and /sapdb/LCName on the shared disks.

Note: /sapdb/LCName includes the kernel, data files and log files for SAP liveCache database server.

When installing SAP liveCache, ensure that the login_id, id_name, group_id, and group_name for the LCNameadm and sdb users is the same on all the nodes.

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

For more details, refer to the SAP product documentation.

- Install the Oracle Application Server instance binaries on shared disks.

- When installing Oracle Application Server, ensure that the login_id, id_name, group_id, and group_name for the Oracle Application Server is the same on all the nodes.

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

The guidelines for installing Oracle Fusion Middleware for high availability are as follows:

- Install the Oracle Fusion Middleware instance binaries on shared disks.

- When installing Oracle Fusion Middleware, ensure that the login_id, id_name, group_id, and group_name for the Oracle Fusion Middleware is the same on all the nodes.

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

About configuring SAP liveCache for high availability

The guidelines for configuring SAP liveCache 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.

Configuring SAP liveCache server for cluster support

Configuring the SAP liveCache server for cluster support involves the following phases:

Installing the SAP liveCache server with database on first node

Install the liveCache server using the SAP-provided installation tool, SAPInst, for the required SAP application.

Start the liveCache installation using `SAPINST_USE_HOSTNAME=VirtualHostName` and perform the following.

- Use the following command to start the installation and proceed as per the instructions on the SAP installation GUI:

```
sapinst SAPINST_USE_HOSTNAME=VirtualHostName
```

- Perform all the post-installation steps described in the SAP liveCache installation guide.

Installing the SAP liveCache server software on second node

Install the liveCache server software on the second node with the same liveCache name used in installing on the first node.

Perform the following steps:

- Create the SAP liveCache database software owner "sdb" user on the system with logon permission disabled with same UID as of first node.

- Create the SAP liveCache database administration group "sdba" on the system with same GID as of the first node. For example on AIX:

  ```
  /usr/bin/mkuser gecos="Database Software Owner" pgrp="sdba" sdb
  /usr/bin/chuser rlogin=false login=false account_locked=true sdb
  ```

- Create the SAP liveCache user "LCName adm" user with same permissions as on first node and with same UID as of first node. For example on AIX:

  ```
  /usr/bin/mkuser gecos="Owner of Database Instance Lc5"
  pgrp="sdba" groups="sapsys" shell="/bin/csh" lc5adm
  ```

- Start the liveCache software installation from the liveCache media with SDBINST.

- Setup the X server logon credentials for all SAP users, namely, LCName adm, control, superdba as described in the SAP note 39439.

- Copy *<LCName>* from the *<indep_data_path>*/config directory on the first node to the *<indep_data_path>*/config directory on the second node.

  For example, `rcp -r /sapdb/data/config/LC5*`
  `vcsaix102:/sapdb/data/config`

  Make sure that all the files are owned by the sdb user and the sdba group.

- Create a work directory for the liveCache server instance
  `<indep_data_path>/wrk/<LCName>`.
  For example,
  ```
  cd /sapdb/data/wrk
  mkdir LC5
  ```

- Create data and log directories for database. The path to the directories should be the same as of the first node.
  For example,
  ```
  cd /sapdb/LC5
  mkdir sapdata
  mkdir saplog
  ```

- Mount the data and log volumes on the first node on to the second node under data and log directories created.

- Start the liveCache server.

Establishing communication between Application server and SAP liveCache server

Each database instance is assigned a logical server name. This logical server name should be used for communication with the database. It is therefore entered in XUSER. As the application instance does not recognize the database server as a local host when the connection is set up, communication via TCP/IP (x_server) is selected. Hence, you need to create a global login for the XUSER entries with virtual hostname on all the nodes. Refer to the SAP note 39439 for more information on how to generate XUSER entries for different users.

Note: If you copy the environment for the *<LCName>* adm user from the first node, then the XUSER.62 file in the HOME directory of the owner, usually *<LCName>*adm, must also be moved and regenerated.

Creating SAP liveCache RFC destination and RFC user

To create liveCache RFC destination and RFC user for SAP Kernel versions that are earlier than version 7.2, follow the instructions on the SAP notes 305634 and 452745.

Create global login for XUSER entries with virtual hostname on all the nodes. See the SAP note 39439 for more information.

Starting, stopping, and initializing SAP liveCache at the operating system level for SAP kernel 7.2 or later

For SAP kernel versions that are earlier than 7.2, the agent for SAP liveCache uses the `startrfc` program to start, stop, and initialize SAP liveCache by using the following function modules:

- START_LIVECACHE_LVC
- STOP_LIVECACHE_LVC
- INITIALIZE_LIVECACHE_LVC.

SAP kernel 7.2 does not include the `startrfc` program in the classic RFC SDK 7.20. Therefore, for SAP kernel 7.2, the SAP liveCache agent cannot use these function modules to start, stop, and initialize SAP liveCache. Instead, the agent uses the `sapevt` program to call the following events to start and stop SAP liveCache:

- Z_PLEASE_START_LIVECACHE
- Z_PLEASE_INITIALIZE_LIVECACHE

■ Z_PLEASE_STOP_LIVECACHE

When any of these events occurs at the operating system level, the system starts the related job, as described in Table 2-1. See the SAP note 568478 for more information.

Table 2-1   Events and their related jobs

Event	Examples of jobs
Z_PLEASE_START_LIVECACHE	START_OF_LIVECACHE_JOB
Z_PLEASE_STOP_LIVECACHE	STOP_OF_LIVECACHE_JOB
Z_PLEASE_INITIALIZE_LIVECACHE	INITIALIZE_OF_LIVECACHE_JOB

To schedule events and their related jobs for the SAP liveCache agent

1   Call transaction SM62 and create the following SAP events:

■ Z_PLEASE_START_LIVECACHE

■ Z_PLEASE_STOP_LIVECACHE

■ Z_PLEASE_INITIALIZE_LIVECACHE

The operating system triggers SAP events by calling the `sapevt` program.

Note: Ensure that the events you create have these exact same names.

2   Call transaction SE38 and create a variant for the respective report (RSLVCSTART, RSLVCSTOP, RSLVCINIT).

Specify only the name of the SAP liveCache, for example, LCA.

RSLVCSTART, RSLVCSTOP, or RSLVCINIT have the same function scope as the respective function module START_LIVECACHE_LVC, STOP_LIVECACHE_LVC, or INITIALIZE_LIVECACHE_LVC.

3   Call transaction SM36, and create new jobs for starting, stopping, and initializing SAP liveCache. For example, START_OF_LIVECACHE_JOB, STOP_OF_LIVECACHE_JOB, and INITIALIZE_OF_LIVECACHE_JOB. Respectively, proceed as follows:

■ On the **Step** tab, enter the report name and the respective variant.

■ On the **Start Condition** tab, select **after event**, and specify the event that you defined in transaction SM62.

Note: For both jobs, select the **periodic job** flag.

To verify that the SAP liveCache instance starts or stops outside of VCS control, run the following commands:

- To start the SAP liveCache instance

  ```
  sapevt Z_PLEASE_START_LIVECACHE name=SAPSID nr=INSTANCE_ID
  ```

- To stop the SAP liveCache instance

  ```
  sapevt Z_PLEASE_STOP_LIVECACHE   name=SAPSID nr=INSTANCE_ID
  ```

Installing, upgrading, and removing the agent for SAP liveCache

This chapter includes the following topics:

Before you install the Cluster Server agent for SAP liveCache

You must install the Cluster Server agent for SAP liveCache on all the systems that will host SAP liveCache service groups.

Before you install the agent for SAP liveCache, 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 27.

- 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 21.

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   *cd1*/aix/vcs/application/acc_library/*version*_library/pkgs

Linux   *cd1*/linux/generic/vcs/application/acc_library/*version*_library/rpms

Solaris   *cd1*/solaris/*dist_arch*/vcs/application/acc_library/*version*_library/pkgs

where *dist_arch* is sol_sparc.

4 If you downloaded the individual ACCLib tar file, navigate to the pkgs 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   `# installp -ac -d VRTSacclib.bff VRTSacclib`

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

Solaris   `# pkgadd -d VRTSacclib.pkg`

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 SAP liveCache 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	`cd1/aix/vcs/application/sap_agent/`
	vcs_version/*version*_agent/pkgs
Linux	`cd1/linux/generic/vcs/application/sap_agent/`
	vcs_version/*version*_agent/rpms
Solaris	`cd1/solaris/`*dist_arch*`/vcs/application/sap_agent/`
	vcs_version/*version*_agent/pkgs
	where, *dist_arch* is sol_sparc

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   `# installp -ac -d`
                `VRTSsaplc.rte.bff VRTSsaplc.rte`

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

Solaris   `# pkgadd -d . VRTSsaplc`

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 31.

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 `VRTSsaplc.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/VRTSsaplc.p5p Veritas`

4   Install the package.

`# pkg install --accept VRTSsaplc`

5   Remove the publisher from the system.

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 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>
```

5 Add a file-based repository in the non-global zone.

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

6 Install the package.

```
# pkg install --accept VRTSsapnw
```

7 Remove the publisher on the non-global zone.

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

8 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
```

9 Enable the publishers that were disabled earlier.

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

Note: Perform steps 2 through 9 on each non-global zone.

Installing the agent in a Solaris 10 brand zone

To install the SAP liveCache 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 . VRTSsaplc
  ```

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

Uninstalling the agent in a VCS environment

You must uninstall the agent for SAP liveCache 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 SAP liveCache resources from the cluster. Run the following command to verify that all resources have been removed:

```
# hares -list Type=SAPLiveCache
```

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

```
# hatype -delete SAPLiveCache
```

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 SAP liveCache from each node in the cluster.

Run the following command to uninstall the agent:

AIX   `# installp -u VRTSsaplc.rte`

Linux   `# rpm -e VRTSsaplc`

Solaris   `# pkgrm VRTSsaplc`

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

```
# pkg uninstall VRTSsapnw
```

For Solaris 11, uninstalling the SAP liveCache agent will also uninstall the SAP NetWeaver agent.

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 in a VCS environment

Perform the following steps to upgrade the agent with minimal disruption, in a VCS environment.

To upgrade the agent in a VCS environment

1 Persistently freeze the service groups that host the application.

```
# hagrp -freeze groupName -persistent
```

2 Stop the cluster services forcibly.

```
# hastop -all -force
```

3 Ensure that the agent operations are stopped on all the nodes.

```
# ps -ef | grep SAPLiveCache
```

4 Uninstall the agent package from all the nodes. Use the platform's native software management program to remove the agent for SAP liveCache from each node in the cluster.

Run the following command to uninstall the agent:

AIX	`# installp -u VRTSsaplc.rte`
Linux	`# rpm -e VRTSsaplc`
Solaris	For Solaris 10: `# pkgrm VRTSsaplc`

5 Install the new agent on all the nodes.

See

6 Copy the new `SAPLiveCacheTypes.cf` file from the agent's conf directory, to the VCS conf directory `/etc/VRTSvcs/conf/config`.

VCS 4.x	■ AIX ■ Linux ■ Solaris	`/etc/VRTSvcs/conf/sample_SAPLiveCache/` `SAPLiveCacheTypes.cf`
VCS 5.x or later	■ AIX ■ Linux	`/etc/VRTSagents/ha/conf/SAPLiveCache/` `SAPLiveCacheTypes.cf`

VCS 5.0	■ Solaris SPARC	/etc/VRTSagents/ha/conf/SAPLiveCache/ SAPLiveCacheTypes50.cf
VCS 5.1 or later	■ Solaris SPARC	/etc/VRTSagents/ha/conf/SAPLiveCache/ SAPLiveCacheTypes51.cf

7 Check for the changes in the resource values required, if any, due to the new agent types file.

Note: To note the list of changed attributes, compare the new type definition file with the old type definition file.

---

8 Start the cluster services.

hastart

9 Start the agent on all nodes, if not started.

haagent -start SAPLiveCache -sys *SystemName*

10 Unfreeze the service groups once all the resources come to an online steady state.

hagrp -unfreeze *GroupName* -persistent

Configuring the agent for SAP liveCache

This chapter includes the following topics:

- About configuring the Cluster Server agent for SAP liveCache
- Importing the agent types files in a VCS environment
- SAP liveCache agent attributes
- Executing a customized monitoring program

About configuring the Cluster Server agent for SAP liveCache

After installing the Cluster Server agent for SAP liveCache, 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 SAP liveCache resources.

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

See "About sample configurations for the agents for SAP liveCache" on page 52.

Importing the agent types files in a VCS environment

To use the agent for SAP liveCache, 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 ■ Linux ■ Solaris	/etc/VRTSvcs/conf/sample_SAPLiveCache/ SAPLiveCacheTypes.cf
VCS 5.x or later	■ AIX ■ Linux	/etc/VRTSagents/ha/conf/SAPLiveCache/ SAPLiveCacheTypes.cf
VCS 5.0	Solaris SPARC	/etc/VRTSagents/ha/conf/SAPLiveCache/ SAPLiveCacheTypes50.cf
VCS 5.1 or later	Solaris SPARC	/etc/VRTSagents/ha/conf/SAPLiveCache/ SAPLiveCacheTypes51.cf

4 Click **Import**.

5 Save the VCS configuration.

The SAP liveCache agent type is now imported to the VCS engine.

You can now create SAP liveCache resources. For additional information about using the VCS GUI, refer to the *Cluster Server Administrator's Guide*.

To import the agent types file using the CLI

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_SAPLiveCache/
	Linux	SAPLiveCacheTypes.cf
	Solaris	
VCS 5.x or later	AIX	/etc/VRTSagents/ha/conf/SAPLiveCache/
	Linux	SAPLiveCacheTypes.cf
VCS 5.0	Solaris SPARC	/etc/VRTSagents/ha/conf/SAPLiveCache/
		SAPLiveCacheTypes50.cf
VCS 5.1 or later	Solaris SPARC	/etc/VRTSagents/ha/conf/SAPLiveCache/
		SAPLiveCacheTypes51.cf

4 Create a dummy `main.cf` file.

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

5 Create the SAP resource type as follows:

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

The SAP liveCache agent type is now imported to the VCS engine. You can now create SAP liveCache resources. For additional information about using the VCS CLI, refer to the *Cluster Server Administrator's Guide*.

SAP liveCache agent attributes

A SAP liveCache service is managed within a VCS environment. VCS uses software agents to control software services within a VCS cluster.

VCS deploys agents to manage all components or resources of the same type. For example, a single mount agent is responsible for managing all mounted file systems that are under VCS control.

The agent attributes define the details that are passed from the VCS engine to the agent to uniquely identify the specific resource component that is to be managed.

Table 4-1 lists the attributes that are required for configuring a SAP liveCache instance.

Table 4-1 Required attributes

Required attributes	Description
DBMPassword	The password for the *DBMUserId* DBM user. The password is stored encrypted and the agent should decrypt appropriately.
	Use the vcsencrypt -agent option to encrypt the password. If you are using VCS GUI to enter password no need to encrypt the password. VCS GUI will automatically encrypt password for you.
	Type and dimension: string-scalar
	Default: No default value
	Example: jxpVmxMpkPlpMpnPo
DBMUserId	The liveCache Database Manager Operator user name used in `dbmcli`" to connect to the liveCache database server.
	Type and dimension: string-scalar
	Example: control.
	Default: "control "
EnvFile	The environments file for *LCUser* to be sourced before starting or stopping the liveCache server. Veritas recommends keeping the environments file on the shared disk for easy maintenance.
	Type and dimension: string-scalar
	Example: `/home/lc5adm/lcenv.csh`
	Default: No default value
InstId	The SAP system number or instance ID for the APO Application Server.
	Type and dimension: integer-scalar
	Example: 01
	Default: No default value
LCHome	The directory where the liveCache database software is installed. This directory is needed to get the `dbmcli` and `x_server` binaries.
	Type and dimension: string-scalar
	Example: `/sapdb/programs/bin`
	Default: No default value

Table 4-1   Required attributes *(continued)*

Required attributes	Description
LCName	The name of the liveCache instance consisting of three alphanumeric characters. Also called liveCache SID. Only upper case letters are allowed and the first letter must be a character. This attribute is needed to uniquely identify the processes of the live Cache server. Also needed by the dbmcli utility to find the status of the liveCache server. Type and dimension: string-scalar Example: LC5. Default: No default value
LCUser	The operating system user for liveCache server. This user is responsible for starting and stopping the liveCache server. It must always be set to a value. For SAP kernel version 7.2 or later, this user name takes the form *SAPSID*adm, whereas for SAP kernel versions earlier than 7.2, this user takes the form *LCName* adm. Type and dimension: string-scalar Example: lc5adm Default: No default value

Table 4-2 lists the optional attributes

Table 4-2 Optional attributes

Optional attribute	Description
ResLogLevel	The string used to set the ResLogLevel of each instance of a resource. This attribute should not be confused with the VCS generic LogLevel Type attribute, which controls the VCS engine log level on a per agent type basis. The ResLogLevel attribute controls the amount of ACCLib VCS Agent Framework based logging that is written to the VCS log file on a per resource -instance basis.
	Valid values are:
	■ ERROR: Only Error level messages are logged.
	■ WARN: Above plus warning level messages are logged.
	■ INFO: Above plus informational level messages will be logged. This is the default log level.
	■ TRACE: Above plus trace level messages will be logged. This is very verbose and should only be used during diagnostic operations.
	Type and dimension: string-scalar
	Example: TRACE
	Default: INFO
	Note: The use of the ResLogLevel attribute is deprecated from VCS version 6.2 onwards. 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. The agent captures the first failure data of the unexpected events and automatically logs debug messages in their respective agent log files.

Table 4-2 Optional attributes *(continued)*

Optional attribute	Description
LogDbg	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.
	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 SAPLiveCache resources in the cluster. If debug logs must be enabled for a specific SAPLiveCache resource, override the LogDbg attribute.
	Type and dimension: string-keylist
	Default: {} (none)
	For more information on how to use the LogDbg attribute, refer to the *Cluster Server Administrator's Guide*.
APOHost	The virtual hostname of the SAP APO application server in which liveCache is configured.
	Type and dimension: string-scalar
	Default: No default value
	Example: sapsc1ci
	Note: For SAP kernel version earlier than 7.2, configure this attribute only if you wish to use the `startrfc` program.
ClientId	The production client ID in which the *RFCDest* and *RFCUserId* are configured for liveCache server. The liveCache server is started and stopped with this client ID.
	Type and dimension: integer-scalar
	Default: No default value
	Example: 001
	Note: For SAP kernel version earlier than 7.2, configure this attribute only if you wish to use the `startrfc` program.

Table 4-2 Optional attributes *(continued)*

Optional attribute	Description
ContainerName	This attribute is used for non-global zone support for VCS 5.0 on Solaris.
	It defines the name of the non-global zone.
	For more details refer to *Cluster Server User's Guide*
	Type and dimension: string-scalar
	Example: saplc-zone1
	Default: No default value
ContainerInfo	This attribute is used for non-global zone support for VCS 5.1 on Solaris.
	It specifies if you can use the service group with the container.
	Assign the following values to the ContainerInfo attribute:
	■ Name: The name of the container. For example, saplc-zone1
	■ Type: The type of container. Set this to Zone.
	■ Enabled: Specify the value as 1, if you want to enable the container. Specify the value as 0, if you want to disable the container.
	For more details refer to the *Cluster Server Administrator's Guide*.
	Type and dimension: string-association
	Example: {Name = saplc-zone1, Type = Zone, Enabled = 1}
	Default: No default value
Language	The language type used for starting and stopping liveCache server. Agent supports only English.
	Type and dimension: string-scalar
	Example: EN
	Default: EN
	Note: For SAP kernel version earlier than 7.2, configure this attribute only if you wish to use the `startrfc` program.

Table 4-2 Optional attributes *(continued)*

Optional attribute	Description
RFCDest	The RFC Destination configured for liveCache server start or stop. In general the destination will be in the form SID CLNT ClientId. Refer to the SAP notes 305634 and 452745 for more information. Type and dimension: string-scalar Example: SC1CLNT001 Default: No default value **Note:** For SAP kernel version earlier than 7.2, configure this attribute only if you wish to use the `startrfc` program.
RFCHome	The home directory for RFC SDK tool kit. `startrfc` binary is used to start or stop the liveCache server with function modules RSLVCSTART or RSLVCSTOP. This RFC tool kit needs to be downloaded from SAP Service Market place from http://service.sap.com/swdc. This directory must have executable permission for LCUser. Type and dimension: string-scalar Example: `/sapdb/rfcsdk/bin` Default: No default value **Note:** For SAP kernel version earlier than 7.2, configure this attribute only if you wish to use the `startrfc` program.
RFCPassword	The password for RFCUserId. ■ store encrypted ■ agent to do decryption appropriately Use the vcsencrypt -agent option to encrypt the password. If you are using VCS GUI to enter password no need to encrypt the password. VCS GUI will automatically encrypt password for you. Type and dimension: string-scalar Example: akgIdkDgbGcgDgeGf Default: No default value **Note:** For SAP kernel version earlier than 7.2, configure this attribute only if you wish to use the `startrfc` program.

Table 4-2   Optional attributes *(continued)*

Optional attribute	Description
RFCUserId	The RFC User configured for SAP liveCache Server to globally operate SAP liveCache Server. Type and dimension: string-scalar Example: LC5RFC Default: No default value **Note:** For SAP kernel version earlier than 7.2, configure this attribute only if you wish to use the startrfc program.
SecondLevelMonitor	Used to enable second-level monitoring and specifies how often it is run. Second-level monitoring is a deeper, more thorough state check of the configured SAP liveCache instance. The numeric value specifies how often that the second-level monitoring routines are run. ■ 0: never run the second-level monitoring routines. ■ 1: run it every monitor interval. ■ 2: run the second-level monitoring routines every second monitor interval, and so on. Care should be taken while setting this attribute to large numbers. For example, if the MonitorInterval is set to 60 seconds, and the SecondLevelMonitor is set to 100, then the dbmcli command for this attribute would only get executed every 100 minutes, which may not be as often as intended. To provide maximum flexibility, the value set is not checked for an upper limit. Hence, you could cause the secondlevelmonitor command to run once a month, if that is what is desired. Type and dimension: integer-scalar Example: 1 Default Value: 0

Table 4-2   Optional attributes *(continued)*

Optional attribute	Description
MonitorProgram	The full path and file name of an external, user-supplied monitor program. If specified, the monitor entry point will execute this file to perform an additional server state check. There are no restrictions for what actions the external monitor program performs to determine the state of a SAP liveCache server. The only constraint is that the external monitor program must return one of the following integer values:
	▪ 0: server is online
	▪ 110: server is online
	▪ 100: server is offline
	▪ 1: server is offline
	▪ 99 or any thing other than{0,1,110,100}: server state is unknown
	Veritas recommends storing the external monitor program on the shared disk directory to ensure the file is always available on the online system. Arguments are supported.
	Type and dimension: string-scalar
	Example1:
	`/sapdb/data/db/wrk/LC5/mymonitor.sh`
	Example2:
	`/sapdb/data/db/wrk/LC5/mymonitor.sh arg1 arg2`
	Default: No default value

Executing a customized monitoring program

The monitor function can execute a customized monitoring utility to perform an additional SAP liveCache server state check.

The monitor function executes the utility specified in the MonitorProgram attribute if the following conditions are satisfied:

▪ The specified utility is a valid executable file.

▪ The process level check and database state check indicates that the SAP liveCache Server instance is online.

The monitor function interprets the utility exit code as follows:

110 or 0   SAP liveCache server instance is online

100 or 1	SAP liveCache server instance is offline
99	SAP liveCache server instance is unknown
Any other value	SAP liveCache server instance is unknown

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

Configuring the service groups for SAP liveCache using the CLI

This chapter includes the following topics:

■ About configuring service groups for SAP liveCache

■ Before configuring the service groups for SAP liveCache

■ Configuring service groups for SAP liveCache

■ Copying the Iccluster file to SAP liveCache directory

About configuring service groups for SAP liveCache

Configuring the SAP liveCache service group involves creating the SAP liveCache 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

See "Configuring service groups for SAP liveCache" on page 44.

Before configuring the service groups for SAP liveCache

Before you configure the SAP liveCache 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 Installation Guide*.

- Verify that SAP liveCache is installed and configured identically on all nodes in the cluster.
  See "About installing SAP liveCache for high availability" on page 14.
  See "About configuring SAP liveCache for high availability " on page 15.

- Verify that the Cluster Server agent for SAP liveCache is installed on all nodes in the cluster.

Configuring service groups for SAP liveCache

While setting up a cluster, you must always ensure that the cluster has some spare capacity to handle the SAP liveCache failover scenarios. For example, in case of a backend database failure, the cluster must be able to run another database instance in conjunction with other running applications.

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 disk groups, disk volumes, file systems, 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.

To add a service group for SAP liveCache server

1 Create a service group for SAP liveCache.

```
# hagrp -add SAP76-LCS
```

For more information on creating a service group refer to the *Cluster Server Administrator's Guide*.

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

```
# hagrp -modify SAP76-LCS SystemList vcssx074 0 vcssx075 1
```

3 Create resources for NIC, IP, DiskGroup, Volume and Mount in the service group.

```
# hares -add SAP76-LCS_nic NIC SAP76-LCS
```

```
# hares -add SAP76-LCS_ip IP SAP76-LCS
```

For more details on creating and modifying resource attributes for NIC, IP, DiskGroup, Volume and Mount refer to the *Cluster Server Bundled Agents Reference Guide*.

4 Create links between the resources.

```
# hares -link SAP76-LCS_ip SAP76-LCS_nic
```

5 Create SAPLiveCache resource for SAP liveCache server.

```
# hares -add SAP76-LCS_lc SAPLiveCache SAP76-LCS
```

Based on the SAP liveCache server instance you are clustering, modify the resource attributes. For more information on agent attributes refer to the

See "SAP liveCache agent attributes" on page 33.

6 Create resource dependencies for SAPLiveCache resource.

The SAPLiveCache resource depends on the IP and Mount resources.

```
# hares -link SAP76-LCS_lc SAP76-LCS_ip
```

7 Verify the final resource dependencies for SAP liveCache server group.

```
# hares -dep
```

For example,

```
    Group         Parent        Child
SAP76-LCS     SAP76-LCS_lc     SAP76-LCS_mnt
SAP76-LCS     SAP76-LCS_lc     SAP76-LCS_ip
SAP76-LCS     SAP76-LCS_ip     SAP76-LCS_nic
SAP76-LCS     SAP76-LCS_mnt    SAP76-LCS_vol
SAP76-LCS     SAP76-LCS_vol    SAP76-LCS_dg
```

Copying the lccluster file to SAP liveCache directory

SAP liveCache server is a cluster aware product. SAP liveCache server provides interfaces to integrate into high availability management system with all liveCache events. This interface is used to prevent the liveCache cluster resource from faulting and failing over for intentional offline of the liveCache system.

Review the information on intentional offline and implementation of the lccluster interface.

See "Intentional offline" on page 56.

See "lccluster implementation for HA management system" on page 56.

To use this feature, the `lccluster` file must be present in the `sap` directory under the SAP liveCache database root directory. Typically, under the `/sapdb/`*LCName*`/db/sap` directory.

Copy the sample lccluster file shipped with the agent to the `sap` directory and set the permissions and owner of the file as follows:

■ Login as superuser.

■ Copy the file to the `sap` directory.
  For VCS 4.1, use the following command

  ```
  cd /opt/VRTSvcs/bin/SAPLiveCache
  ```

  For VCS 5.0, use the following command

  ```
  cd /opt/VRTSagents/ha/bin/SAPLiveCache
  ```

  ```
  cp lccluster /sapdb/LCName/db/sap
  ```

■ Change the owner of the file with the command.

  ```
  cd /sapdb/LCName/db/sap
  chown sdb:sdba lccluster
  ```

■ Change the permissions of the file with the command.

  ```
  chmod 555 lccluster
  ```

Troubleshooting the agent for SAP liveCache

This chapter includes the following topics:

- Using the correct software and operating system versions
- Meeting prerequisites
- Configuring SAP liveCache resources
- Starting the SAP liveCache 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 SAP liveCache supports, see the Veritas Services and Operations Readiness Tools (SORT) site: https://sort.veritas.com/agents.

Meeting prerequisites

Before installing the agent for SAP liveCache, ensure that the following prerequisites are met.

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

Configuring SAP liveCache resources

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

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

You can then restart the SAP liveCache 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 SAP instance outside the cluster framework, is described as follows.

To restart the SAP liveCache server instance outside the framework

1 Log in as a superuser.

2 Ensure that the SAP database is up and running. Refer to the relevant database documentation or consult your database administrator for more information.

3 Use the LCUser attribute to log in to the SAPliveCache server:

```
# su LCUser
$ USER=LCUser; LOGNAME=LCUser; HOME=/home/LCUser
$ export USER LOGNAME HOME
$ . EnvFile
```

For certain shell versions on AIX, LOGNAME is read-only.

4   Start the SAP liveCache server instance:

For SAP kernel version 7.2 or later:

```
sapevt Z_PLEASE_START_LIVECACHE name=SAPSID nr=INSTANCE_ID
```

For SAP kernel versions that are earlier than 7.2:

```
$ Path_RFCSDK/bin/startrfc -3 -d RFCDest -h APOHost -s InstId
-c ClientId -u RFCserId -p RFCPassword -1 Language
-F START_LIVECACHE_LVC -E IV_CON_NAME=LCA
```

5   Ensure that the SAP liveCache instance is running successfully by running the grep command for LCName.

For example, for an SAP liveCache server instance LC5:

```
$ ps -ef | grep LC5
```

You must see the kernel processes running on the system for liveCache server.

If the SAP liveCache server instance is running outside the cluster framework, you can then attempt to restart the SAP liveCache server within the cluster framework by enabling the liveCache resource.

Reviewing error log files

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

Using SAP liveCache log files

If a SAP liveCache server is facing problems, you can access the server log files to further diagnose the problem. The SAP liveCache log files are located in the /sapdb/data/wrk/<LCName> directory.

Reviewing cluster log files

In case of problems while using the agent for SAP liveCache, 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`.

Additionally, you can also refer to the latest SAPLiveCache agent log files located at `/var/VRTSvcs/log/SAPLiveCache_A.log`

Note: Include both these log files while addressing the problem to Veritas support team.

Using trace level logging

The ResLogLevel attribute controls the level of logging that is written in a cluster log file for each SAP liveCache 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.`

Warning: You may consider temporarily increasing the timeout values for SAPLiveCache for debugging purposes. After the debugging process is complete, you can revert back to the original timeout values.

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 SAPLiveCache

◆ Enable the debug log.

```
# hatype -modify SAPLiveCache 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 SAPLiveCache LogDbg
# hares -modify SAPLiveCache LogDbg DBG_5
```

Sample Configurations

This appendix includes the following topics:

- About sample configurations for the agents for SAP liveCache

- Sample agent type definition for SAP liveCache

- Sample SAP liveCache resource configuration for VCS

- Sample service group configurations for SAP liveCache

- Intentional offline

- lccluster implementation for HA management system

About sample configurations for the agents for SAP liveCache

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 SAP liveCache. For more information about these resource types, refer to the *Cluster Server Bundled Agents Reference Guide*.

Sample agent type definition for SAP liveCache

Examples of agent type definition files follow.

VCS 4.1

```
type SAPLiveCache (
    static str ArgList[] = { ResLogLevel, State, IState, EnvFile,
    LCHome, LCUser, LCName, DBMUserId, DBMPassword, APOHost, InstID,
```

```
    ClientID, RFCDest, RFCHome, RFCUserId, RFCPassword, Language,
    SecondLevelMonitor, MonitorProgram }
    str ResLogLevel = INFO
    str EnvFile
    str LCHome
    str LCUser
    str LCName
    str DBMUserId = control
    str DBMPassword
    str APOHost
    int InstId
    int ClientId
    str RFCDest
    str RFCHome
    str RFCUserId
    str RFCPassword
    str Language = EN
    int SecondLevelMonitor = 0
    str MonitorProgram
)
```

VCS 5.0

```
type SAPLiveCache (
        static str AgentDirectory =
"/opt/VRTSagents/ha/bin/SAPLiveCache"
        static str AgentFile = "/opt/VRTSvcs/bin/Script50Agent"

static str ArgList[] = { ResLogLevel, State, IState, EnvFile,
LCHome, LCUser, LCName, DBMUserId, DBMPassword, APOHost, InstId,
ClientId, RFCDest, RFCHome, RFCUserId, RFCPassword, Language,
SecondLevelMonitor, MonitorProgram }
        str ResLogLevel = INFO
        str EnvFile
        str LCHome
        str LCUser
        str LCName
        str DBMUserId = control
        str DBMPassword
        str APOHost
        str InstId
        str ClientId
        str RFCDest
        str RFCHome
```

```
            str RFCUserId
            str RFCPassword
            str Language = EN
            int SecondLevelMonitor = 0
            str MonitorProgram
    )
```

Sample SAP liveCache resource configuration for VCS

The sample resource configuration for VCS is as follows:

```
SAPLiveCache SAPLC5_liveCache (

ResLogLevel        = "INFO"
EnvFile            = "/home/lc5adm/lcenv.csh"
LCHome             = "/sapdb/programs/bin"
LCUser             = "lc5adm"
LCName             = "LC5"
DBMUserId          = "control"
DBMPassword        = "asfjfdsyuklf"
APOHost            = "sapsc1ci"
InstId             = 01
ClientId           = 001
RFCDest            = "SC1CLNT001"
RFCHome            = "/sapdb/rfcsdk/bin"
RFCUserId          = "LC5RFC"
RFCPassword        = "SaskDWlksa"
Language           = "EN"
SecondLevelMonitor = 0
MonitorProgram     = ""

    )
```

Sample service group configurations for SAP liveCache

Create SAP liveCache resource using VCS GUI. The following diagrams provide the sample configuration of VCS resource for SAP liveCache Server.

illustrates the dependency graph for the configuration.
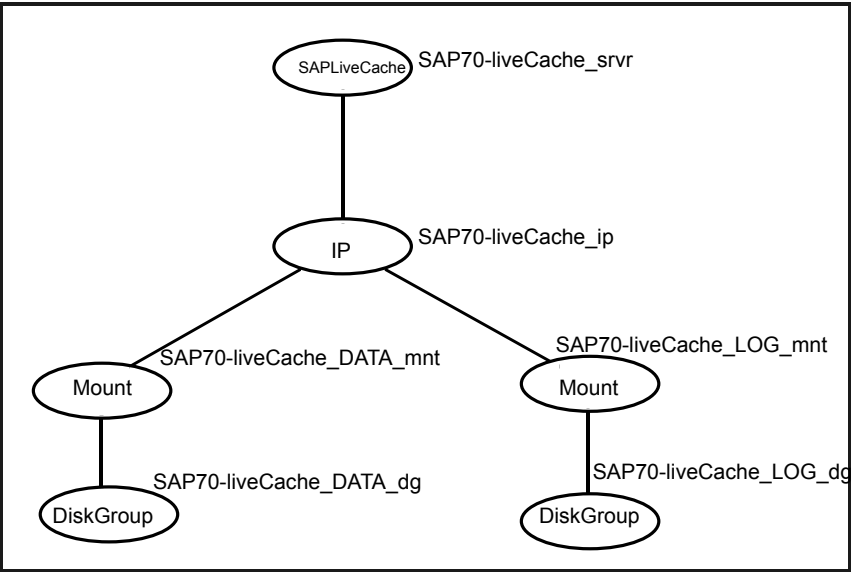
Figure A-1 Dependency graph for SAP liveCache

Table A-1 illustrates the agent attributes for SAP liveCache.

Table A-1 Agent attributes for SAP liveCache

Attribute	Value
ResLogLevel	INFO
EnvFile	/home/lc3adm/.lcenv.csh
LCHome	/sapdb/programs/bin
LCUser	lc3adm
LCName	HOT
DBMUserId	control
DBMPassword	govSjsLshSukKuiQvmHmh
APOHost	saplc1ci
InstId	11
ClientId	001
RFCDest	LC1CLNT001

Table A-1   Agent attributes for SAP liveCache *(continued)*

Attribute	Value
RFCHome	/usr/sap/rfcsdk/bin
RFCUserId	LC3RFC
RFCPassword	FLTkrITILghk
Language	EN
SecondLevelMonitor	0
MonitorProgram	

Intentional offline

Intentional offline is needed to detect graceful shutdown of the SAP liveCache server with liveCache transaction LC10. SAP liveCache server has a mechanism to notify the high availability management system whenever the liveCache server is started or stopped gracefully with the help of lccluster. The lccluster script must be implemented if the HA management system needs to be notified for all graceful actions.

All graceful starts of the liveCache server can be detected by the SAP liveCache agent without the help of the lccluster script. For defined MonitorInterval, monitor entry point is run by the VCS HA management system for all the resources of type SAPliveCache. Hence, the agent can easily detect whenever SAP liveCache server is started gracefully with maximum time delay of MonitorInterval.

Generally, for all graceful stops of the SAP liveCache server, the agent faults the resource assuming unexpected offline. The Clean entry point will be called and if the resource is set to "critical", the service group will be failed-over to the other node in the cluster, which is not intended. Hence, to prevent this scenario, intentional offline has been implemented.

lccluster implementation for HA management system

SAP recommends starting/stopping the liveCache server for APO with liveCache transaction LC10. The transaction LC10 calls SAP provided script /sapdb/<LCName>/db/sap/lcinit to start/stop the liveCache instance. The lcinit script

can inform the high availability management system with the start/stop actions of the liveCache with the help of another script /sapdb/LCName/db/sap/lccluster, if implemented. Otherwise lcinit assumes that the liveCache is running in a non-HA environment. In case of SAP liveCache starts and stops in HA environments, lcinit calls lccluster with the following options:

starting req	lcinit begins to start the liveCache
starting ok	lcinit has started the liveCache successfully
starting error	start up of the liveCache failed
stopping req	lcinit begins to stop the liveCache
stopping ok	lcinit has stopped the liveCache successfully
stopping error	stop of the liveCache failed

This script must be implemented to prevent faulting of SAP liveCache server resource for intentional stop of liveCache server with transaction LC10.

This script is implemented such that whenever a liveCache is started/stopped out of the liveCache transaction LC10, the lccluster script, creates a tmp file with the name symc_lccluster_LCName under the liveCache working directory /sapdb/data/wrk/LCName.

Get the LCName from lccluster script full path name.

Following is a sample code for lccluster script.

```
bash-2.05b$ cat /sapdb/LC5/db/sap/lccluster
#!/opt/VRTSperl/bin/perl -w
use Fcntl;
use File::Path qw ( mkpath );
my $sClustMode  = $ARGV[0];
my $sStatus          = $ARGV[1];
my $sClustStatus  = "$sClustMode $sStatus";
my ( $sFileName, $sSAP, $sDB, $sLCName ) = reverse $0 =~
m@([^/]+)@g;
my $sWrkDir   = "/sapdb/data/wrk/" . $sLCName ;
my $sClustFile = $sWrkDir . "/symc_lccluster_" . $sLCName ;
#-----------------------------
# Create work folder if needed..
#-----------------------------
if ( -d $sWrkDir ) {
  # Directory already exists.
}
```

```
elsif ( ! mkpath ( $sWrkDir, 0, 0700 ) ) {
  # Not able to create a directory.
  return 0;
}
if ( sysopen( CLUSTFILE, $sClustFile,
O_WRONLY|O_SYNC|O_CREAT|O_TRUNC, 0660 ) ) {
  print CLUSTFILE $sClustStatus;
}
else {
  # Not able to create the file.
}
close ( CLUSTFILE );
bash-2.05b$
```