Oracle® Solaris Cluster Data Service for Oracle Web Tier Guide

Part No: E39659
July 2014, E39659–01

ORACLE

Contents

Using This Documentation

- **Overview** – Explains how to install and configure the Oracle Solaris Cluster HA for Oracle Web Tier (HA for Oracle Web Tier) data service.
- **Audience** – Experienced system administrators with extensive knowledge of Oracle software and hardware.
- **Required knowledge** – Knowledge of the Oracle Solaris operating system, of Oracle Solaris Cluster software, and expertise with the volume manager software that is used with Oracle Solaris Cluster software.

This document is not to be used as a planning or presales guide.

Product Documentation Library

Late-breaking information and known issues for this product are included in the documentation library at http://www.oracle.com/pls/topic/lookup?ctx=E39579.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info or visit http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs if you are hearing impaired.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program web site at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Feedback

Provide feedback about this documentation at http://www.oracle.com/goto/docfeedback.

♦ ♦ ♦ CHAPTER 1

Installing and Configuring HA for Oracle Web Tier

This chapter describes the steps to install and configure Oracle Solaris Cluster HA for Oracle Web Tier (HA for Oracle Web Tier) on your Oracle Solaris Cluster servers.

This chapter contains the following sections:

You can configure HA for Oracle Web Tier as a failover data service in the global zone or in a zone cluster. See Chapter 1, "Planning for Oracle Solaris Cluster Data Services," in "Oracle Solaris Cluster Data Services Planning and Administration Guide " and the "Oracle Solaris Cluster Concepts Guide " document for an overview of failover and scalable data services.

Planning the Installation and Configuration

Have available the following information before you install HA for Oracle Web Tier:

- The path to the application binaries. You can install the binaries on the local disks, on a highly available local file system, on the cluster file system, or on a Network Attached Storage (NAS) device. See "Configuration Guidelines for Oracle Solaris Cluster Data Services" in "Oracle Solaris Cluster Data Services Planning and Administration Guide " for a discussion of the advantages and disadvantages of each location.
- Whether to host the Oracle Web Tier `htdocs` subdirectory on a highly available local file system, a cluster file system, or a NAS device.
- The names of the resource groups and resources you will create.

- The names of the nodes that will master the data service. The nodes can be physical or virtual machines or the nodes that form a zone cluster.
- The logical hostname that clients use to access the data service. You typically set up this IP address when you install the cluster. See the "Oracle Solaris Cluster Concepts Guide " for details on network resources.

Overview of the Installation and Configuration Process for HA for Oracle Web Tier

The table below lists the sections that describe the installation and configuration tasks.

TABLE 1-1 Task Map: Installing and Configuring HA for Oracle Web Tier

Task	Instructions
Install and configure the Oracle Web Tier software	"Installing and Configuring Oracle Web Tier Software" on page 10
Install the HA for Oracle Web Tier packages	"Installing the HA for Oracle Web Tier Package" on page 16
Configure and start HA for Oracle Web Tier	"Registering and Configuring HA for Oracle Web Tier Components" on page 18
Tune the HA for Oracle Web Tier fault monitor	"Tuning the HA for Oracle Web Tier Fault Monitors" on page 24

Installing and Configuring Oracle Web Tier Software

To install the Oracle Web Tier software, perform the following procedure.

See Oracle Web Tier documentation for standard installation instructions. See `Oracle Solaris Cluster 4 Compatibility Guide (http://www.oracle.com/technetwork/server-storage/solaris-cluster/overview/solariscluster4-compatibilityguide-1429037.pdf)` for information about Oracle Web Tier versions that are supported with the Oracle Solaris Cluster software.

▼ How to Install and Configure the Oracle Web Tier Software and Resources

To install the Oracle Web Tier software, you must first create your chosen shared storage and logical host resources. Once complete, you must install an Oracle Fusion Middleware (FMW) environment before you install and configure the Oracle Web Tier software.

When you configure the Oracle Web Tier components as a failover service, you place the Oracle Process Management and Notification Server (OPMN) resource, the Oracle HTTP Server resource, and the network resource in a single resource group.

When a highly available local file system is used to store either the `htdocs` software or the `htdocs` and Oracle Web Tier software, then an HAStoragePlus storage resource is also placed in the failover resource group. Alternatively, when a cluster file system is used in a zone cluster, or when a NAS device is used, then a storage resource is placed in a separate scalable resource group.

Before You Begin

- Verify that all the network addresses that you use have been added to your name-service database.

  You should have performed this verification during your initial Oracle Solaris Cluster installation. See the planning chapter in the "Oracle Solaris Cluster Software Installation Guide " for details.

Note - To avoid failures because of name-service lookup, verify that all the network addresses are present in the `/etc/inet/hosts` file on all of the cluster nodes. Configure the name service mapping by using the `svccfg -s svc:/system/name-service/switch` command. Ensure that the `config/host` entry is configured to first check the local files before accessing NIS, NIS+, or DNS.

- Ensure that the `/etc/netmasks` file has IP-address subnet and netmask entries for all logical hostnames. If necessary, edit the `/etc/netmasks` file to add any missing entries.

1. **Create an Oracle Solaris user account to own the Oracle Fusion Middleware and Oracle Web Tier software.**

   Perform this step on all nodes that host the service. The user and group IDs for the account must be consistent on all nodes.

2. **Create a failover resource group to hold the network and application resources.**

   This resource group contains both network and failover application resources. If a highly available local file system is used to store the `htdocs` and application binaries directories, then the resource group also contains an HAStoragePlus storage resource.

   Optionally, you can specify with the `-n` option the set of nodes on which the data service can run.

   ```
   # clresourcegroup create [-n node-zone-list] resource-group
   ```

   resource-group

   Specifies the name of the failover resource group to add. This name can be your choice but must be unique for the resource groups within the cluster.

[-n *node-list*]

Specifies a comma-separated, ordered list of nodes that can master this resource group.

This list is optional. If you omit this list, the global zone of each cluster node can master the resource group.

3. Bring the failover resource group online.

```
# clresourcegroup online -M resource-group
```

resource-group

Specifies the name of the failover resource group.

4. Add a logical hostname network resource to the failover resource group that you created in Step 2.

This is the hostname that clients will use to connect to the Oracle HTTP Server component.

```
# clreslogicalhostname create -g resource-group \
-h hostname[,…] [-N netiflist] resource
```

-h *hostname,…*

Specifies a comma-separated list of network resources to add.

resource-group

Specifies the name of the failover resource group that you created in Step 2.

resource

Specifies a resource name. If you do not supply your choice for a resource name, the name of the network resource defaults to the first name that is specified after the -h option.

-N *netiflist*

Specifies an optional, comma-separated list that identifies the IPMP groups that are on each node. The format of each entry in the list is *netif*@*node*. The replaceable items in this format are as follows:

netif	Specifies an IPMP group name, such as `sc_ipmp0`, or a public network interface card (NIC). If you specify a public NIC, Oracle Solaris Cluster attempts to create the required IPMP groups.
node	Specifies the name or ID of a node.

Note - If you require a fully qualified hostname, you must specify the fully qualified name with the -h option and you cannot use the fully qualified form in the resource name.

Note - Oracle Solaris Cluster does not currently support using the adapter name for `netif`.

5. **Configure the storage resources.**

- **If you are storing the Oracle HTTP Server `htdocs` directory on a highly available local file system or on a cluster file system in the global zone, create a storage resource in the application failover resource group.**

```
# clresource create -g resource-group \
-t SUNW.HAStoragePlus \
{-p FileSystemMountPoints=mount-point,… | -p Zpools=zpool,...} \
resource
```

resource-group

Specifies the name of the failover resource group that you created in Step 2.

`-p FileSystemMountPoints` *mount-point-list*

Specifies a comma-separated list of file system mount points to add.

`-p Zpools` *zpool-list*

Specifies a comma-separated list of zpools to add.

resource

Specifies a resource name.

- **If you are storing the Oracle HTTP Server `htdocs` and application binaries directories on a NAS mounted file system or on a cluster file system in a zone cluster, create a scalable resource group to contain the storage resource.**

a. **Create a scalable resource group to contain the storage resource.**

```
# clresourcegroup create [-n node-zone-list] \
-p Maximum_primaries=m \
-p Desired_primaries=n \
resource-group
```

resource-group

Specifies the name of the scalable service resource group to add.

`-p Maximum_primaries=`*m*

Specifies the maximum number of active primary nodes allowed for this resource group. If you do not assign a value to this property, the default is `1`.

`-p Desired_primaries=`*n*

Specifies the desired number of active primary nodes allowed for this resource group. If you do not assign a value to this property, the default is `1`.

`-n` *node-list*

Specifies a comma-separated, ordered list of nodes that can master this resource group. The format of each entry in the list is *node*.

This list is optional. If you omit this list, the global zone of each cluster node can master the resource group.

b. Add a storage resource to the scalable resource group that you created in Step 5.2.a.

For example, an HAStoragePlus resource to manage a cluster file system that is to be mounted in a zone cluster is added as follows:

```
# clresource create -g resource-group \
-t SUNW.HAStoragePlus \
-p FileSystemMountPoints=mount-point \
resource
```

resource-group

Specifies the name of the scalable resource group that you created in Step 5.2.a.

`-p FileSystemMountPoints` *mount-point*

Specifies a comma-separated list of file system mount points to add.

resource

Specifies a resource name.

c. Bring the scalable resource group online.

```
# clresourcegroup online -M resource-group
```

resource-group

Specifies the name of the scalable resource group.

6. As the software owner created in Step 1, use the Oracle Fusion Middleware 11*g* installation media to install the Oracle WebLogic Server software.

Note - Do not yet run `quickstart` or configure the software.

---

7. As the software owner, use the Oracle Web Tier installation media to install the software only.

a. **When prompted, run the `createCentralInventory.sh` script to create the central inventory.**

b. **If the software has been installed centrally, do the following:**

i. **When the installation is complete, switch the resource group containing the software to the other nodes that can host the service.**

ii. **Rerun the `createCentralInventory.sh` script to create the required directories in the `/var` hierarchy on the other nodes.**

8. **If necessary, update the Oracle Web Tier software with the latest Service Repository Updates (SRUs) and security fixes.**

9. **As the software owner, create an Oracle WebLogic Administration Server component for the domain.**

a. **To ensure that the Enterprise Manager GUI uses the logical hostname, set the following environment variables.**

```
$ export LD_PRELOAD_32=$LD_PRELOAD_32:/usr/cluster/lib/libschost.so.1
$ export LD_PRELOAD_64=$LD_PRELOAD_64:/usr/cluster/lib/64/libschost.so.1
$ export SC_LHOSTNAME=logical-hostname
```

b. **Create the component.**

```
$ cd /FMW-Home/Web-Tier-Home/common/bin
$ ./config.sh
```

10. **If you want to make the Oracle WebLogic Administration Server component highly available, create the necessary resource groups and resources to control it.**

For more details on how to perform this step, see “Oracle Solaris Cluster Data Service for Oracle WebLogic Server Guide ”.

11. **As the software owner, create the Oracle HTTP Server component instance.**

a. **To ensure that the Enterprise Manager GUI uses the logical hostname, set the following environment variables.**

```
$ export LD_PRELOAD_32=$LD_PRELOAD_32:/usr/cluster/lib/libschost.so.1
$ export LD_PRELOAD_64=$LD_PRELOAD_64:/usr/cluster/lib/64/libschost.so.1
$ export SC_LHOSTNAME=logical-hostname
```

b. **Create the component instance.**

```
$ cd /FMW-Home/Web-Tier-Home/bin
$ ./config.sh
```

Note - You can create both HTTP and Web Cache instances, but the HA for Oracle Web Tier only currently supports the HTTP instances. Observe the following requirements for HTTP instances:

- When you are prompted to provide the instance name and the instance subdirectory, the instance name and the final component of the instance subdirectory must be the same. For example, if the Oracle Web Tier instance is named *myinst1*, then the corresponding directory for that instance must be of the form */FMW-Home/Web-Tier-Home*`/instances/`*myinst1*.
- The Oracle HTTP Server instance name must be the same on all nodes.

Failure to meet these requirements will result in validation failures when attempting to create an Oracle HTTP Server resource.

12. Manually stop the Oracle HTTP Server and Oracle Process Management and Notification Server instances.

```
$ cd /FMW-Home/Web-Tier-Home/instances/Instance-Name/bin
$ ./opmnctl stopall
```

13. Edit the `http.conf` file.

```
$ cd /FMW-Home/Web-Tier-Home/instances/Instance-Name/config/OHS/Component-Instance-Name
$ vi http.conf
```

Ensure that the file contains a line of the following form, where *Logical-Hostname* is the logical hostname resource that you created in Step 4:

`Listen` *Logical-Hostname*:*Port*

See Also The following sections provide examples of this procedure, as used in conjunction with the procedure to register and configure HA for Oracle Web Tier component resources:

- Example 1-1
- Example 1-2

Installing the HA for Oracle Web Tier Package

If you did not install the HA for Oracle Web Tier package during your initial Oracle Solaris Cluster installation, perform this procedure to install the package.

How to Install the HA for Oracle Web Tier Package

Perform this procedure on each cluster node where you want the HA for Oracle Web Tier software to run.

1. **On the cluster node where you are installing the data service package, assume the `root` role.**

2. **Ensure that the data service package is available from the configured publisher and that the `solaris` and `ha-cluster` publishers are valid.**

```
# pkg list -a ha-cluster/data-service/oracle-http-server ha-cluster/data-service/oracle-pmn-server
# pkg publisher
PUBLISHER                             TYPE     STATUS   P  LOCATION
solaris                               origin   online   F  solaris-repository
ha-cluster                            origin   online   F  ha-cluster-repository
```

For information about setting the `solaris` publisher, see "Adding, Modifying, or Removing Package Publishers" in "Adding and Updating Software in Oracle Solaris 11.2".

Tip - Use the `-nv` options whenever you install or update to see what changes will be made, such as which versions of which packages will be installed or updated and whether a new BE will be created.

If you do not get any error messages when you use the `-nv` options, run the command again without the `-n` option to actually perform the installation or update. If you do get error messages, run the command again with more `-v` options (for example, `-nvv`) or more of the package FMRI pattern to get more information to help you diagnose and fix the problem. For troubleshooting information, see Appendix A, "Troubleshooting Package Installation and Update," in "Adding and Updating Software in Oracle Solaris 11.2".

3. **Install the HA for Oracle Web Tier software package.**

```
# pkg install ha-cluster/data-service/oracle-http-server \
ha-cluster/data-service/oracle-pmn-server
```

4. **Verify that the package installed successfully.**

```
$ pkg info ha-cluster/data-service/oracle-http-server \
ha-cluster/data-service/oracle-pmn-server
```

Installation is successful if output shows that `State` is `Installed`.

5. **Perform any necessary updates to the Oracle Solaris Cluster software.**

For instructions on updating your software, see Chapter 11, "Updating Your Software," in "Oracle Solaris Cluster System Administration Guide".

Registering and Configuring HA for Oracle Web Tier Components

The sections that follow contain instructions for registering and configuring the HA for Oracle Web Tier component resources. For information about the extension properties, see Appendix A, "HA for Oracle Web Tier Extension Properties". The Tunable entry indicates when you can update a property.

See the `rt_properties`(5), `r_properties`(5), and `rg_properties`(5) man pages for details on all of the Oracle Solaris Cluster extension properties.

To set an extension property of a resource, include the following option in the `clresource` command that creates or modifies the resource:

`-p` *property*=*value*

`-p` *property*

Identifies the extension property that you are setting.

value

Specifies the value to which you are setting the extension property.

You can also use the procedures in Chapter 2, "Administering Data Service Resources," in "Oracle Solaris Cluster Data Services Planning and Administration Guide " to configure resources after the resources are created.

Tools for Registering and Configuring HA for Oracle Web Tier

Oracle Solaris Cluster provides the following tools for registering and configuring the HA for Oracle Web Tier components:

- **Oracle Solaris Cluster maintenance commands.** For more information, see "How to Register and Configure HA for Oracle Web Tier *by Using Oracle Solaris Cluster (CLI)*" on page 18.

▼ How to Register and Configure HA for Oracle Web Tier *by Using Oracle Solaris Cluster (CLI)*

Complete the registration and configuration on any cluster member.

1. **On a cluster member, assume the `root` role or assume a role that provides `solaris.cluster.admin` and `solaris.cluster.modify` RBAC authorization.**

2. **Register the `ORCL.ohs` and `ORCL.opmn` resource types for the data service.**

   ```
   # clresourcetype register ORCL.ohs ORCL.opmn
   ```

3. **Create the application resources in the application failover resource group.**
   You must create two resources in the application failover resource group, one for the Oracle Process Management and Notification Server component and one for the Oracle HTTP Server component.

   a. **Create the Oracle Process Management and Notification Server resource.**

   ```
   # clresource create -g resource-group
   -t ORCL.opmn \
   -p Oracle_home=ORACLE_HOME-path \
   -p Instance_Name=instance-name \
   [-p Debug_level=debug-level] \
   [-p Resource_project_name=project-name] \
   [-p Resource_dependencies_offline_restart=offline-restart-resource-dependencies[,…]] \
   resource
   ```

   resource-group

   Specifies the name of the failover resource group created in Step 2 of "How to Install and Configure the Oracle Web Tier Software and Resources".

   `-p Oracle_home=`*ORACLE_HOME-path*

   Specifies the absolute path to the Oracle Web Tier software installation. This is normally a subdirectory of the Oracle Fusion Middleware installation.

   `-p Instance_Name=`*instance-name*

   Specifies the OPMN instance name that was supplied when the Oracle Web Tier software was configured. The default value is `instance1`.

   `-p Debug_level=`*debug-level*

   An optional property that specifies the amount of debugging information produced by the resource and fault probe. The default value is `0` which results in no debugging output.

   `-p Resource_project_name=`*project-name*

   An optional property that specifies the project under which the OPMN resource should run. The default value is the project called `default`.

   All processes that are started by the OPMN resource inherit this project setting. This includes any Oracle HTTP server that is started by OPMN. Consequently, for any OHS resource that is dependent on an OPMN resource, the value of the

`Resource_project_name` property that is set for the OHS resource must be identical to the value that is set for the OPMN resource or left as the default.

`-p Resource_dependencies_offline_restart=`*resource-dependencies-offline-restart*[,...]

An optional property that specifies a comma-separated list of resources on which this resource has an offline restart dependency. This list must include a dependency on the storage resource created in Step 5 of "How to Install and Configure the Oracle Web Tier Software and Resources", with the exception of configurations where a cluster file system is used in the global zone. Furthermore, the storage resource dependency must have `{local_node}` scope.

b. Create the Oracle HTTP Server (OHS) resource.

```
# clresource create -g resource-group \
-t ORCL.ohs \
-p Component_instance=component-instance-name \
[-p Debug_level=debug-level] \
-p Resource_dependencies=ohs-lh \
-p Resource_dependencies_offline_restart=opmn-rs \
resource
```

resource-group

Specifies the name of the failover resource group created in Step 2 of "How to Install and Configure the Oracle Web Tier Software and Resources".

`-p Component_instance=`*component-instance-name*

Specifies the name of the Oracle HTTP Server component instance that is under the control of the Oracle Process Management and Notification Server component configured in Step 3.a. The default value is `ohs1`.

`-p Debug_level=`*debug-level*

An optional property that specifies the amount of debugging information produced by the resource and fault probe. The default value is `0` which results in no debugging output.

`-p Resource_dependencies=`*ohs-lh*

Specifies the logical hostname resource created in Step 4 of "How to Install and Configure the Oracle Web Tier Software and Resources", on which this resource has a dependency.

`-p Resource_dependencies_offline_restart=`*opmn-rs*

Specifies a dependency with `{local_node}` scope on the Oracle Process Management and Notification Server resource created in Step 3.a, on which this resource has an offline-restart dependency.

Example 1-1 Registering HA for Oracle Web Tier on a Highly Available Local File System

This example shows how to register an Oracle Web Tier service that uses a highly available local file system on a two-node cluster. The following are the sample names used in the commands:

Node names	`phys-schost-1, phys-schost-2`
Zpool name (for highly available local file systems)	`ohspool`
Logical hostname	`schost-1`
Resource group (for all of the resources)	`ohs-rg`
Logical hostname resource	`ohs-lh-rs`
HAStoragePlus storage resource	`ohs-hasp-rs`
Oracle HTTP Server component resource	`ohs-rs`
Oracle Process Management and Notification Server component resource	`opmn-rs`

Add a failover resource group to contain all of the resources

```
# clresourcegroup create ohs-rg
```

Bring the failover resource group online

```
# clresourcegroup online -M ohs-rg
```

Add the logical hostname resource to the failover resource group

```
# clreslogicalhostname create -g ohs-rg -h schost-1 ohs-lh-rs
```

Register the HAStoragePlus resource type, if it is not already registered

```
# clresourcetype register SUNW.HAStoragePlus
```

Register the Oracle HTTP Server resource type

```
# clresourcetype register ORCL.ohs
```

Register the Oracle Process Management and Notification Server resource type

```
# clresourcetype register ORCL.opmn
```

Add the HAStoragePlus resource to the failover resource group

```
# clresource create -g ohs-rg \
-t SUNW.HAStoragePlus -p Zpools=ohspool ohs-hasp-rs
```

Install and configure the Oracle Web Tier software

Add the Oracle Process Management and Notification Server component resource to the failover resource group

```
# clresource create -g ohs-rg \
-t ORCL.opmn -p Oracle_home=/ohspool/Oracle/Middleware/Oracle_WT1 \
-p Instance_name=myinstance \
-p Resource_dependencies_offline_restart=ohs-hasp-rs{local_node} opmn-rs
```

Add the Oracle HTTP Server component resource to the failover resource group

```
# clresource create -g ohs-rg \
-t ORCL.ohs -p Component_instance=myohs \
-p Resource_dependencies=ohs-lh-rs \
-p Resource_dependencies_offline_restart=opmn-rs{local_node} ohs-rs
```

Example 1-2 Registering HA for Oracle Web Tier on a Cluster File System

This example shows how to register an Oracle Web Tier service that uses a cluster file system in a zone cluster on a two-node cluster. The commands are run in the zone cluster and the zone cluster has been granted access to the logical host and cluster file system required. The following are the sample names used in the commands:

Node names	`zchost-1, zchost-2`
Cluster file system	`/global/ohs`
Logical hostname	`zchost-lh-1`
Resource group for the failover resources	`ohs-rg`
Resource group for the cluster file system resource	`stor-rg`
Logical hostname resource	`ohs-lh-rs`
HAStoragePlus storage resource	`ohs-hasp-rs`
Oracle HTTP Server component resource	`ohs-rs`

Oracle Process Management and Notification Server component resource	`opmn-rs`

Add a failover resource group to contain the failover resources

```
# clresourcegroup create ohs-rg
```

Bring the failover resource group online

```
# clresourcegroup online -M ohs-rg
```

Add a scalable resource group to contain the storage resource

```
# clresourcegroup create -S -p Maximum_primaries=2 -p Desired_primaries=2 stor-rg
```

Register the HAStoragePlus resource type, if it is not already registered

```
# clresourcetype register SUNW.HAStoragePlus
```

Add the HAStoragePlus resource to the scalable resource group

```
# clresource create -g stor-rg \
-t SUNW.HAStoragePlus -p FileSystemMountPoints=/global/ohs ohs-hasp-rs
```

Bring the scalable resource group online

```
# clresourcegroup online -M stor-rg
```

Add the logical hostname resource to the failover resource group

```
# clreslogicalhostname create -g ohs-rg -h zchost-lh-1 ohs-lh-rs
```

Register the Oracle HTTP Server resource type

```
# clresourcetype register ORCL.ohs
```

Register the Oracle Process Management and Notification Server resource type

```
# clresourcetype register ORCL.opmn
```

Install and configure the Oracle Web Tier software

Add the Oracle Process Management and Notification Server component resource to the failover resource group

```
# clresource create -g ohs-rg \
-t ORCL.opmn -p Oracle_home=/ohspool/Oracle/Middleware/Oracle_WT1 \
-p Instance_name=myinstance \
-p Resource_dependencies_offline_restart=ohs-hasp-rs{local_node} opmn-rs
```

Add the Oracle HTTP Server component resource to the failover resource group

```
# clresource create -g ohs-rg \
-t ORCL.ohs -p Component_instance=myohs \
-p Resource_dependencies=ohs-lh-rs \
-p Resource_dependencies_offline_restart=opmn-rs{local_node} ohs-rs
```

How to Verify Data Service Installation and Configuration

After you configure HA for Oracle Web Tier, verify that you can open a web page with the network resources (logical hostname) and port number from a web browser. Perform a switchover with the `clresourcegroup` command to verify that the service continues to run on a secondary node and can be switched back to the original primary.

Tuning the HA for Oracle Web Tier Fault Monitors

The HA for Oracle Web Tier fault monitors are contained in the resources whose resource types are `ORCL.ohs` and `ORCL.opmn`.

System properties and extension properties of the resource control the behavior of the fault monitor. The default values of these properties determine the default behavior of the fault monitor. The default behavior should be suitable for most Oracle Solaris Cluster installations. Therefore, you should tune the HA for Oracle Web Tier fault monitors *only* if you need to modify this default behavior.

Tuning the HA for Oracle Web Tier fault monitors involves the following tasks:

- Setting the interval between fault monitor probes
- Setting the timeout for fault monitor probes
- Defining the criteria for persistent faults
- Specifying the failover behavior of a resource

Information about the HA for Oracle Web Tier fault monitor that you need to perform these tasks is provided in the subsections that follow.

Tune the HA for Oracle Web Tier fault monitor when you register and configure HA for Oracle Web Tier or after initial configuration. For more information, see “Registering and Configuring HA for Oracle Web Tier Components” on page 18.

Updates to the `probe_timeout`, `start_timeout`, `stop_timeout`, and `thorough_probe_interval` properties result in comparable updates in the `opmn.xml` file.

For detailed information, see “Tuning Fault Monitors for Oracle Solaris Cluster Data Services” in “Oracle Solaris Cluster Data Services Planning and Administration Guide ”.

Operations by the HA for Oracle Web Tier Fault Monitors

The two resource types, `ORCL.ohs` and `ORCL.opmn`, contain separate fault probes that query the health of the Oracle HTTP Server and Oracle Process Management and Notification Server components, respectively.

Operations by the Oracle Process Management and Notification Server Fault Monitor

The `ORCL.opmn` fault probe for the Oracle Process Management and Notification Server component performs the following steps:

- Checks that the `opmnctl` command exists in the */ORACLE-HOME*`/instances/`*INSTANCE-NAME*`/bin` directory, and that the script is executable.
- Checks that the `opmn.xml` file is valid by using the following command:

  ```
  $ opmnctl validate
  ```

- If either of these two checks fail, then an attempt is made to fail over (give over) the service to another node.
- If both checks succeed, then the command `opmnctl ping` is run.
  - If this command succeeds, the resource status is set to `OK` and the probe returns with an exit code of `0`.
  - If this command fails, the resource status is set to `FAULTED` and the probe returns with an exit code of `100`, causing the resource to attempt to restart.

Operations by the Oracle HTTP Server Fault Monitor

Because the Oracle HTTP Server component is under the control of Oracle Process Management and Notification Server component, the `ORCL.opmn` fault probe obtains the status of the Oracle HTTP Server component from the Oracle Process Management and Notification Server component. This is done in two stages:

- Checks that an Oracle HTTP Server component with type `OHS` is found in the output of the following command:

  ```
  $ opmnctl status ias-component=COMPONENT-INSTANCE -noheaders -fmt "%typ"
  ```

- Checks that the Oracle HTTP Server component is reported as `ALIVE` by the following command:

  ```
  $ opmnctl status ias-component=COMPONENT-INSTANCE -noheaders -fmt "%sta"
  ```

If the fault probe is successful, the resource status is set to `OK` and the probe returns with an exit code of `0`. If the fault probe fails, the resource status is set to `FAULTED` and the probe returns with an exit code of `100`, causing the resource to attempt to restart.

Note - If the Oracle HTTP Server component is used as a load-balancer through the `mod_wl_ohs` plugin, then the Oracle Process Management and Notification Server component can declare that the Oracle HTTP Server component is `DOWN` if none of the load-balancing targets are available. In these circumstances, the fault probe for the Oracle HTTP Server component attempts to restart the service. You can avoid such behavior by creating a dependency between the load-balancer resource and the target resources.

Actions in Response to Faults

Based on the history of failures, a failure can cause either a local restart or a failover of the data service. For detailed information, see “Tuning Fault Monitors for Oracle Solaris Cluster Data Services” in “Oracle Solaris Cluster Data Services Planning and Administration Guide ”.

Upgrading the HA for Oracle Web Tier Resource Types

Upgrade the `ORCL.ohs` and `ORCL.ohs` resource types if the following conditions apply:

- You are upgrading from an earlier version of the HA for Oracle Web Tier data service.
- You need to use the new features of this data service.

For general instructions that explain how to upgrade a resource type, see “Upgrading a Resource Type” in “Oracle Solaris Cluster Data Services Planning and Administration Guide ”. The information that you require to complete the upgrade of the `ORCL.ohs` and `ORCL.ohs` resource types is provided in the subsections that follow.

Information for Registering the New Resource Type Version

The relationship between a resource type version and the release of Oracle Solaris Cluster data services is shown in the following table. The release of Oracle Solaris Cluster data services indicates the release in which the version of the resource type was introduced.

Resource Type Version	Oracle Solaris Cluster Data Services Release
1	4.0

To determine the version of the resource type that is registered, use the `clresourcetype show` command.

The resource type registration (RTR) files for the `ORCL.ohs` and `ORCL.opmn` resource types are `/opt/ORCLscohs/etc/ORCL.ohs` and `/opt/ORCLscopmn/etc/ORCL.opmn`, respectively.

Information for Migrating Existing Instances of the Resource Type

The information that you require to edit each instance of the `ORCL.ohs` or `ORCL.opmn`resource type is as follows:

- You can perform the migration at any time.
- If you need to use the features of a newer version of the HA for Oracle Web Tier data service, the required value of the `Type_version` property will be greater than that of the initial release, that is `1`.

The following example shows a command for modifying an instance of the `ORCL.ohs` resource type.

EXAMPLE 1-3 Migrating Instances of the `ORCL.ohs` Resource Type

```
# clresource set -p Type_version=2 ohs-rs
```

This command modifies the `ORCL.ohs` resource named `ohs-rs` as follows:

- The `Type_version` property of this resource is set to the value of a (hypothetical) newer release `2`.

♦ ♦ ♦ APPENDIX A

HA for Oracle Web Tier Extension Properties

The following sections describe the extension properties for the following resource types:

- "ORCL.ohs Extension Properties" on page 29
- "ORCL.opmn Extension Properties" on page 32

These resource types represent the two components of the Oracle Web Tier application in a Oracle Solaris Cluster configuration.

For details about system-defined properties, see the r_properties(5) man page and the rg_properties(5) man page.

ORCL.ohs Extension Properties

The extension properties of the ORCL.ohs resource type are as follows:

Child_mon_level

Child monitoring level for the process monitoring facility (PMF). This property is inherited from the SUNW.gds resource type and should not be changed.

Data type Integer

Default –1

Tunable When disabled.

Component_instance

The name of the Oracle HTTP Server instance listed in the opmnctl output.

Data type String

Default ohs1

Tunable When disabled.

`Debug_level`

Determines the amount of debug information produced.

Data type Integer

Default `0`

Per node `True`

Range `0 – 2`

Tunable At any time.

`Failover_enabled`

Determines whether to failover when `retry_count` is exceeded during `retry_interval`.

Data type Boolean

Default `True`

Tunable When disabled.

`Log_level`

Determines the log level for event based traces.

Data type `Enum`

Default `NONE`

Range `NONE`, `INFO`, or `ERR`

Tunable At any time.

`Monitor_retry_count`

The number of times that the process monitor facility (PMF) restarts the fault monitor during the time window that the `Monitor_retry_interval` property specifies. This property refers to restarts of the fault monitor itself rather than to the resource. The system-defined properties `Retry_interval` and `Retry_count` control restarting of the resource.

Data type Integer

Default `4`

Range 0 - 2147483647

-1 indicates an infinite number of retry attempts.

Tunable At any time

`Monitor_retry_interval`

The time (in minutes) over which failures of the fault monitor are counted. If the number of times that the fault monitor fails exceeds the value that is specified in the extension property `Monitor_retry_count` within this period, the PMF does not restart the fault monitor.

Data type Integer

Default 2

Range 0 – 2147483647

-1 indicates an infinite retry interval.

Tunable At any time

`Network_aware`

Determines whether the application uses network. This property is inherited from the `SUNW.gds` resource type and should not be changed.

Data type Boolean

Default False

Tunable At creation.

`Probe_timeout`

The timeout value (in seconds) that the fault monitor uses to probe the resource.

Data type Integer

Default 90

Range 0 – 2147483641

Tunable At any time.

`Stop_signal`

The signal sent to the application for being stopped.

Data type Integer

Default 15 (SIGTERM)

Range 1 (SIGHUP) to 37 (SIGLOST)

Tunable When disabled.

ORCL.opmn Extension Properties

Child_mon_level

Child monitoring level for the process monitoring facility (PMF). This property is inherited from the SUNW.gds resource type and should not be changed.

Data type Integer

Default –1

Tunable When disabled.

Debug_level

Determines the amount of debug information produced.

Data type Integer

Default 0

Per-node True

Range 0 - 2

Tunable At any time.

Failover_enabled

Determines whether to failover when retry_count is exceeded during retry_interval.

Data type Boolean

Default True

Tunable When disabled.

`Instance_name`

The instance name. A directory of this name must exist within the `ORACLE_HOME/instances` subdirectory.

Data type String

Default `instance1`

Tunable When disabled.

`Log_level`

Determines the log level for event based traces.

Data type Enum

Default `NONE`

Range `NONE`, `INFO`, or `ERR`

Tunable At any time.

`Monitor_retry_count`

The number of times that the process monitor facility (PMF) restarts the fault monitor during the time window that the `Monitor_retry_interval` property specifies. This property refers to restarts of the fault monitor itself rather than to the resource. The system-defined properties `Retry_interval` and `Retry_count` control restarting of the resource.

Data type Integer

Default `4`

Range `0` - `2147483647`

`–1` indicates an infinite number of retry attempts.

Tunable At any time

`Monitor_retry_interval`

The time (in minutes) over which failures of the fault monitor are counted. If the number of times that the fault monitor fails exceeds the value that is specified in the extension property `Monitor_retry_count` within this period, the PMF does not restart the fault monitor.

Data type Integer

Default `2`

Range 0 – 2147483647

-1 indicates an infinite retry interval.

Tunable At any time

`Network_aware`

Determines whether the application uses network. This property is inherited from the SUNW.gds resource type and should not be changed.

Data type Boolean

Default False

Tunable At creation.

`Oracle_home`

The absolute path of the ORACLE_HOME of the Oracle Web Tier component of Oracle Fusion Middleware.

Data type String

Default No default value.

Tunable When disabled.

`Probe_timeout`

The timeout value (in seconds) that the fault monitor uses to probe the resource.

Data type Integer

Default 90

Range 0 – 2147483641

Tunable At any time.

`Stop_signal`

The signal sent to the application for being stopped.

Data type Integer

Default 15 (SIGTERM)

Range 1 (SIGHUP) to 37 (SIGLOST)

Tunable When disabled.

Index

T

V

W