

Cluster Server 7.3

Application Note: Dynamic Reconfiguration for Oracle Servers - Solaris

Last updated: 2017-05-31

Legal Notice

Copyright © 2017 Veritas Technologies LLC. All rights reserved.

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

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

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

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

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

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

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

<http://www.veritas.com>

Technical Support

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

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

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

<https://my.veritas.com>

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

Worldwide (except Japan)

CustomerCare@veritas.com

Japan

CustomerCare_Japan@veritas.com

Documentation

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

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

Documentation feedback

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

doc.feedback@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

Dynamic reconfiguration of Oracle servers

This document includes the following topics:

- [Overview: Dynamic reconfiguration in a VCS environment](#)
- [Supported software and hardware](#)
- [Preparing to perform dynamic reconfiguration](#)
- [Scenarios requiring a VCS shutdown](#)
- [Stopping and starting VCS](#)
- [Performing dynamic reconfiguration on Oracle SunFire \(s6800; e12K/15K/ e25K\)](#)
- [Replacing an online Host Bus Adapter \(HBA\) on an M5000 server](#)

Overview: Dynamic reconfiguration in a VCS environment

This application note describes how to perform Dynamic Reconfiguration operations on VCS clustered system domains of the Oracle TM servers.

The dynamic reconfiguration operations typically include configuring and unconfiguring CPU/memory boards to and from domains and configuring and unconfiguring I/O boards in a domain. These operations allow switching boards from one domain to another or permit removing a board or card to upgrade or replace it. Dynamic reconfiguration operations can be performed while the operating environment continues to run.

However, a dynamic reconfiguration operation performed on a CPU/memory board that has permanent memory requires that the system domain be temporarily

suspended. In this case, VCS must be stopped. Do not use the following procedures to dynamically reconfigure a system board containing a VCS private heartbeat link. If you need to do so, you must stop VCS before proceeding.

For a dynamic reconfiguration operation performed on an I/O board, ensure that all devices that are in use and belong to the I/O board are released, i.e., they are not in use by any application modules.

For users of Storage Foundation for Oracle RAC, it is necessary to stop the Oracle RAC instance within the domain being reconfigured if VCS must be stopped. This permits communications among other RAC instances to occur while the instance in the one domain is temporarily stopped.

See [“Scenarios requiring a VCS shutdown”](#) on page 6.

See [“Stopping and starting VCS”](#) on page 7.

Boards with I/O controllers can be dynamically reconfigured as long as you use VxVM with the Dynamic Multi-Pathing (DMP) feature to manage the shared storage.

The Solaris dynamic reconfiguration utility enables you to reconfigure the resources of system boards so that the boards can be replaced without system downtime.

In such cases, before you can physically remove a board, you must “detach” it, or reconfigure it such that its resources can be disabled and removed from the domain configuration. Likewise, after you have physically replaced a board in a domain, you must “attach” it, or reconfigure it into the domain.

The Oracle documentation for dynamic reconfiguration contains comprehensive descriptions of procedures and commands. To avoid damaging system boards and components, you should be familiar with the procedures for their removal and replacement.

Note: Currently, VCS does not support using dynamic reconfiguration in clusters where I/O controllers and storage use Multiplexed I/O (MPxIO).

Supported software and hardware

Following is a list of supported software and hardware requirements:

Supported software

For latest information on supported software, refer to *Veritas InfoScale Release Notes*.

Supported hardware

- Oracle SunFire/Enterprise servers (s6800, e12K/15K, e10K, e25K)

Preparing to perform dynamic reconfiguration

Make sure that you determine which devices on the system board will be impacted by the dynamic reconfiguration operations and determine how to mitigate the impact.

To be dynamically reconfigured, the boards must satisfy the following conditions:

- Critical resources on the boards must be redundant. For example, boards for which CPUs and memory are redundant can be reconfigured after their function has been replaced and their activity stopped. A CPU board that contains the only CPU in a domain cannot be moved.
- A memory board containing permanent memory, such as the OpenBoot™ PROM or kernel memory, can be moved after the memory has been moved to another board. Dynamic reconfiguration on boards with permanent memory requires VCS to be shut down.
- Disk drives must be accessible via alternate pathways. The Dynamic Multi-Pathing (DMP) feature can provide alternate paths. Before moving a host bus adapter (HBA), switch all the card's functions to an alternate card. An HBA that controls sole access to an active drive cannot be moved.
- Activity on a PCI card must be stopped before the card is removed.

Scenarios requiring a VCS shutdown

It is necessary to stop VCS and unconfigure GAB and LLT in certain circumstances.

VCS must be shutdown under the following circumstances:

- When performing dynamic reconfiguration on a system board (CPU/Memory board) with permanent memory.
- When the I/O board requiring reconfiguration contains all the private network links used by the domain.
- When the I/O board contains the only public network links used by the domain.
- When the I/O board contains all of the paths to a storage device.

The necessity of performing a VCS shutdown can be reduced by some device layout planning before clustering the domains.

CPU/Memory boards with permanent memory

If the CPU/memory board to be removed contains permanent memory, the operating system's function must be suspended to permit dynamic reconfiguration to occur. In such a case, VCS must be stopped.

However, you do not need to stop VCS when you are performing dynamic reconfiguration on a board that does not contain permanent memory. Typically, in a domain with multiple CPU/memory boards, one board has permanent memory, while the others do not. When you are performing dynamic reconfiguration to add a new board to the domain, the existing functions in the domain are not affected by the dynamic addition of a new CPU/memory board.

Note: If you must reconfigure multiple boards and a board with permanent memory is among them, reconfigure the board with permanent memory last. This sequence ensures minimum VCS downtime.

To determine if the CPU/memory board has permanent memory

- 1 Log in to the domain as domain administrator.
- 2 List the boards with permanent memory in the domain by entering the following command:

```
# cfgadm -av | grep permanent
```

```
SB2::memory connected configured ok base address 0x1e000000000,  
16777216 KBytes total, 2001200 KBytes permanent
```

The output in the example shows SB2 to contain permanent memory. Before this board can be dynamically reconfigured, VCS must be stopped.

See [“Stopping and starting VCS”](#) on page 7.

Other CPU/memory boards in the domain do not contain permanent memory and may be dynamically reconfigured without stopping VCS.

Stopping and starting VCS

This section contains the procedures for stopping VCS if it is required for dynamic reconfiguration and the procedures for starting VCS if it has been stopped for dynamic reconfiguration.

- See [“Stopping VCS in a standard environment”](#) on page 8.
- See [“Restarting VCS in a standard environment”](#) on page 10.

- See [“Stopping VCS in SF for Oracle RAC environment”](#) on page 11.
- See [“Restarting VCS in SF for Oracle RAC environment”](#) on page 14.

Stopping VCS in a standard environment

When you dynamically reconfigure CPU/Memory boards and I/O boards, it may be necessary, in some circumstances, to stop VCS in the domain.

Applications running on clusters of three or more domains remain highly available on two or more domains if VCS operation must be stopped on one domain. In a cluster of two domains, the applications running during reconfiguration are not highly available when VCS must be stopped on one of the domains.

If you are running SF for Oracle RAC, see [Stopping VCS in SF for Oracle RAC environment](#)

To stop VCS in a standard environment

- 1 Log in as administrator to the domain (dom1, for example) you are reconfiguring.
- 2 List the VCS service groups to determine which are online on the domain.

```
# hagrpl -list
```

- 3 If you can switch the service groups running on the domain to another domain (dom2, for example), switch the service groups.

```
# hagrpl -switch service_grp_name -to dom2
```

Verify that the service groups are offline on dom1.

```
# hastatus
```

Stop VCS on dom1.

```
# hstop -local
```


- 4** If you cannot switch the online service groups to another system, freeze each of them for the duration of dynamic reconfiguration.

Make the VCS configuration writable.

```
# haconf -makerw
```

Freeze each of the service groups persistently.

```
# hagrps -freeze service_grp_name -persistent
```

Verify the groups are frozen.

```
# hagrps -display | grep Frozen
```

Make the configuration read-only.

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

Stop VCS.

```
# hastop -local -force
```

- 5** Unconfigure I/O fencing module

```
# /sbin/vxfenconfig -U
```

- 6** Unconfigure GAB.

```
# /sbin/gabconfig -U
```

- 7** Unconfigure LLT.

```
# /sbin/lltconfig -U
```

Answer “y” to confirm that you want to stop LLT.

- 8** Stop VxFen, GAB, and LLT modules if required.

```
# svcadm disable -t system/vxfen
```

```
# svcadm disable -t system/gab
```

```
# svcadm disable -t system/llt
```

9 Remove the VxFen, GAB, and LLT modules from the kernel.

Determine the IDs of the VxFen, GAB, and LLT modules:

```
# modinfo | egrep "vxfen|gab|llt"

305 78531900 30e 305 1 gab
292 78493850 30e 292 1 llt
```

Unload the GAB and LLT modules based on their module IDs:

```
# modunload -i 305
# modunload -i 292
```

10 You can begin performing dynamic reconfiguration.

Restarting VCS in a standard environment

If you are ready to restart VCS in the domain where you are performing dynamic reconfiguration, use the following procedure. If you are running SF for Oracle RAC, and are ready to restart VCS, see [Restarting VCS in SF for Oracle RAC environment](#).

To restart LLT, GAB, and VCS**1** Restart LLT.

```
# svcadm enable system/llt
```

2 Restart GAB.

```
# svcadm enable system/gab
```

3 Restart I/O fencing.

```
# svcadm enable system/vxfen
```

4 Start VCS.

```
# hastart
```

5 Verify GAB and VCS are started.

```
# /sbin/gabconfig -a
```

```
GAB Port Memberships
```

```
=====
```

```
Port a gen 4a1c0001 membership 012
```

```
Port h gen g8ty0002 membership 012
```

To bring service groups online

- 1 Determine which service groups are frozen.

```
# hagrps -display | grep Frozen
```

- 2 Make the configuration writable.

```
# haconf -makerw
```

- 3 Unfreeze the frozen service groups.

```
# hagrps -unfreeze service_grp_name -persistent
```

- 4 Make the configuration read-only.

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

Stopping VCS in SF for Oracle RAC environment

If you must stop VCS on a domain where SF for Oracle RAC is running, the Oracle RAC application on the domain being reconfigured must be brought offline. In addition, the GAB, LLT, LMX, and VXFEN modules must be unconfigured. Performing these steps ensures that other instances do not attempt communication with the stopped instance. This could cause the application to hang, when the instance does not respond.

You need to perform these prerequisites before you stop VCS in SF for Oracle RAC environment.

Note: You need to log in as administrator into the domain that needs to be reconfigured.

Stopping Oracle Instances outside VCS control

- ◆ If the database is not managed by VCS, stop the Oracle database using the following command:

```
# srvctl stop database -d db_name
```

Stopping the applications that use CVM or CFS (outside of VCS control)

- 1 Stop the applications that use CVM volumes or CFS mount points that are not controlled by VCS, using native application commands.
- 2 Verify that no processes use the CFS mount point using the following command:

```
# fuser -c mount_point
```

Unmounting CFS file systems (outside of VCS control)

You need to unmount the CFS file systems that are not under VCS control on all nodes.

- 1 Determine the file systems that need to be unmounted by verifying the output of the mount command. The command lists the mounted clustered file systems.

Note: Check the main.cf file for identifying the files that are under VCS control.

```
# mount -v | grep vxfs | grep cluster
```

- 2 Unmount each file system that are not controlled by VCS using the following command:

```
# umount mount_point
```

To stop VCS in a SF for Oracle RAC environment

- 1 Log in as administrator to the domain being reconfigured (dom1, for example).
- 2 List the configured VCS service groups and see which are online in the domain:

```
# hagrpl -list
```

- 3 Based on the output of step 2, bring each service group that is online to offline in the domain dom1. Use the following command:

```
# hagrpl -offline service_grp_name -sys dom1
```

- 4 Stop VCS.

```
# hastop -local
```

In addition to port h, this command stops the CVM drivers using ports v and w.

- 5 If any CFS file systems outside of VCS control are mounted, unmount them.
- 6 Stop ODM using the following command:

```
# svcadm disable -t vxodm
```

- 7 Unconfigure the VCSMM and I/O fencing drivers, which use ports b and o, respectively:

```
# /sbin/vxfenconfig -U
```

```
# /sbin/vcsmmconfig -U
```

- 8 Unconfigure the LMX driver:

```
# /sbin/lmxconfig -U
```

- 9** Verify that the drivers h, v, w, f, q, d, b, and o are stopped. They should not show memberships when you use the `gabconfig -a` command:

```
# gabconfig -a

GAB Port Memberships
=====
Port a gen 4a1c0001 membership 01
```

- 10** Stop cluster fencing, VCSMM, LMX, ODM, and GAB modules if required.

```
# svcadm disable -t system/vxfen
# svcadm disable -t system/vcssmm
# svcadm disable -t system/vxodm
# svcadm disable -t system/gab
# svcadm disable -t system/llt
```

- 11** Unload the VCSMM, I/O fencing, and LMX modules.

Determine the module IDs for VCSMM, I/O fencing, and LMX:

```
# modinfo | egrep "lmx|vxfen|vcssmm"

237 783e4000 25497 237 1 vcssmm (VERITAS Membership
Manager)
238 78440000 263df 238 1 vxfen (VERITAS I/O Fencing)
239 7845a000 12b1e 239 1 lmx (LLT Mux 3.5B2)
```

Unload the VCSMM, I/O fencing, and LMX modules based on their module IDs:

```
# modunload -i 237
# modunload -i 238
# modunload -i 239
```

- 12** Unconfigure GAB

```
# /sbin/gabconfig -U
```

- 13** Unconfigure LLT

```
# /sbin/lltconfig -U
```

14 Remove the GAB and LLT modules from the kernel.

Determine the IDs of the GAB and LLT modules:

```
# modinfo | egrep "gab|llt"

305 78531900 30e 305 1 gab
292 78493850 30e 292 1 llt
```

Unload the GAB and LLT modules based on their module IDs:

```
# modunload -i 305
# modunload -i 292
```

15 You can begin performing dynamic reconfiguration.

Restarting VCS in SF for Oracle RAC environment

If you used the procedure described in [Stopping VCS in SF for Oracle RAC environment](#) before dynamically reconfiguring a CPU/memory board, use the following procedures to restart VCS and bring the service groups on the domain online.

To restart LLT, GAB, VCS, and DBE/AC processes**1** Restart LLT.

```
# svcadm enable system/llt
```

2 Restart GAB.

```
# svcadm enable system/gab
```

3 Restart the LMX driver.

```
# svcadm enable system/lmx
```

4 Restart the VCSMM driver.

```
# svcadm enable system/vcsmm
```

5 Restart the VXFEN driver

```
# svcadm enable system/vxfen
```

6 Restart the ODM driver.

```
# svcadm enable system/odm
```

7 Start VCS.

```
# hastart
```

- 8** Verify that the CVM service group is online.

```
# hagr -state cvm
```

- 9** Verify the GAB memberships required for DBE/AC for Oracle9i RAC are configured.

```
# /sbin/gabconfig -a
```

```
GAB Port Memberships
=====
Port a gen 4a1c0001 membership 012
Port b gen g8ty0002 membership 012
Port d gen 40100001 membership 012
Port f gen f1990002 membership 012
Port h gen g8ty0002 membership 012
Port o gen f1100002 membership 012
Port q gen 28d10002 membership 012
Port v gen 1fc60002 membership 012
Port w gen 15ba0002 membership 012
```

- 10** Bring the service groups that had been take offline in See 3 on page 8.

```
# hagr -online service_grp_name -sys dom1
```

- 11** Start the Oracle instances that are outside VCS control using the following commands:

```
# srvctl start database -d db_name
```

Mount the CFS files systems and start the application using the native application commands that are not managed by VCS.

Performing dynamic reconfiguration on Oracle SunFire (s6800; e12K/15K/ e25K)

You may dynamically reconfigure CPU/memory boards, I/O boards and PCI on I/O boards for Oracle SunFire s6800/e12K/e15K/e25K.

- See [“Performing dynamic reconfiguration on a CPU/memory board”](#) on page 16.
- See [“Performing dynamic reconfiguration on PCI cards on I/O boards”](#) on page 22.
- See [“Performing dynamic reconfiguration on I/O boards”](#) on page 25.

Performing dynamic reconfiguration on a CPU/memory board

You may want to remove a CPU/memory board that is malfunctioning or you may want to reconfigure a board from one domain to another where it is needed more.

To reassign a board from one domain to another, you must unconfigure it from one domain and reassign it to another domain. This can be done without physically removing the board from its slot. To replace a board, however, you must unconfigure it from one domain, physically remove it, add its replacement board and reconfigure it to the domain.

Use the following procedures to dynamically reconfigure a CPU/memory board.

To determine the status of the board you are reconfiguring

- 1 If necessary, log in as the administrator to the domain containing the CPU/memory board.
- 2 Determine the attachment point of the board you are removing:

```
# cfgadm

Ap_Id Type Receptable Occupant Cond
.
N0.SB2 CPU connected configured ok
.
```

- 3 Make sure you have checked whether the board has permanent memory.

See [“To determine if the CPU/memory board has permanent memory”](#) on page 7.

- If the board in the domain you want to dynamically reconfigure contains permanent memory, be sure you have first stopped VCS using the procedures described in See [“Stopping and starting VCS”](#) on page 7.
 - See [“Stopping VCS in a standard environment”](#) on page 8.
 - See [“Restarting VCS in a standard environment”](#) on page 10.
 - See [“Stopping VCS in SF for Oracle RAC environment”](#) on page 11.
 - See [“Restarting VCS in SF for Oracle RAC environment”](#) on page 14.
- If the board you want to reconfigure does not contain permanent memory, you can proceed to dynamically reconfigure it.

To unbind processes bound to CPU on the board

- 1 To determine if any processes are bound to a CPU, enter:

```
# pbind -q
```

- 2 If a processes is bound to the board, the output indicates the process ID and the ID number of the CPU.

```
process id 650: 0
```

- 3 If you see no output or see output showing no processes bound to a CPU on the board, you are reconfiguring, perform the steps in [To unconfigure the board](#).

- 4 Unbind all processes bound to the CPU on the board. For example, enter:

```
# pbind -u 650
```

- 5 Rebind the processes to a processor on another board, if necessary. For example, bind process 650 to processor with ID 9, which is on another board, using the command:

```
# pbind -b 650 9
```

- 6 If you attempt to unconfigure a board with processes bound to it, you receive a message that resembles:

```
cfgadm: Hardware specific failure: unconfigure SB15: Failed to  
off-line:dr@0:SB15::cpu3
```

To unconfigure the board

1 Unconfigure and disconnect the board:

```
# cfgadm -v -c disconnect SB2
```

2 If the board does not contain permanent memory, the command's output resembles the following with slight variations for each server:

```
request delete capacity (4 cpus)
request delete capacity (2097152 pages)
request delete capacity SB2 done
request offline SUNW_cpu/cpu448
request offline SUNW_cpu/cpu449
request offline SUNW_cpu/cpu450
request offline SUNW_cpu/cpu451
request offline SUNW_cpu/cpu448 done
request offline SUNW_cpu/cpu449 done
request offline SUNW_cpu/cpu450 done
request offline SUNW_cpu/cpu451 done
unconfigure SB2
unconfigure SB2 done
notify remove SUNW_cpu/cpu448
notify remove SUNW_cpu/cpu449
notify remove SUNW_cpu/cpu450
notify remove SUNW_cpu/cpu451
notify remove SUNW_cpu/cpu448 done
notify remove SUNW_cpu/cpu449 done
notify remove SUNW_cpu/cpu450 done
notify remove SUNW_cpu/cpu451 done
disconnect SB2
disconnect SB2 done
poweroff SB2
poweroff SB2 done
unassign SB2 skipped
```

Skip to [4](#).

3 If the board has permanent memory, the system prompts you to proceed:

```
System may be temporarily suspended; proceed (yes/no)?
```

If the answer is “yes,” dynamic reconfiguration proceeds. The system is suspended during reconfiguration. When the system resumes operation on another board, the board you are reconfiguring is disconnected. If the disconnect operation succeeds, the output resembles the following with slight variations for different servers:

```
request suspend SUNW_OS
request suspend SUNW_OS done
request delete capacity (2097152 pages)
request delete capacity SB15 done
request offline SUNW_cpu/cpu480
request offline SUNW_cpu/cpu481
request offline SUNW_cpu/cpu482
request offline SUNW_cpu/cpu483
request offline SUNW_cpu/cpu480 done
request offline SUNW_cpu/cpu481 done
request offline SUNW_cpu/cpu482 done
request offline SUNW_cpu/cpu483 done
unconfigure SB15
unconfigure SB15 done
notify remove SUNW_cpu/cpu480
notify remove SUNW_cpu/cpu481
notify remove SUNW_cpu/cpu482
notify remove SUNW_cpu/cpu483
notify remove SUNW_cpu/cpu480 done
notify remove SUNW_cpu/cpu481 done
notify remove SUNW_cpu/cpu482 done
notify remove SUNW_cpu/cpu483 done
disconnect SB15
disconnect SB15 done
poweroff SB15
```

Skip to [4](#).

Note: If there are real-time processes running on the board you are unconfiguring, the disconnect operation may not succeed. You must stop these processes in the appropriate manner before continuing with dynamic reconfiguration.

- 4** If the board has real-time processes that must be stopped, the dynamic reconfiguration operation fails, indicating the PID of those processes that are running. There may be slight variations in output for different Oracle Sun Enterprise servers.

For example:

```
.
.
notify remove SUNW_cpu/cpu481 done
notify remove SUNW_cpu/cpu482 done
notify remove SUNW_cpu/cpu483 done
cfgadm: Hardware specific failure: unconfigure SB15:
Cannot
quiesce realtime thread: 621
```

- 5** To determine the name of the processes, use the command:

```
# ps -ef | grep PID
```

- 6** Stop the process in the appropriate manner. For example, the processes in our example must be stopped using the kill command:

```
# kill -9 PID
```

- 7** Retry the command in [1](#).

- 8** To verify the board is disconnected and unconfigured, use the `cfgadm` command:

```
# cfgadm

Ap_Id Type Receptable Occupant Cond
.
N0.SB2 CPU disconnected unconfigured unknown
.
```

Now you can remove the board from the slot, or reassign it to another domain.

Note: Do not remove the board until you have verified it is disconnected.

- 9** If you are replacing the board immediately, see [To add a board to a domain](#). Otherwise, return the cluster to operation without replacing the disconnected CPU/memory board using the procedure in the following section.

To add a board to a domain

- 1 Log in as administrator to the domain where you plan to add or configure the boards.
- 2 If you are adding a new or a replacement board to a domain (for example, dom1), verify the state of the slot to contain the board.

To be configured with a new board, the slot must have the following states and condition:

- Receptacle state: empty
- Occupant state: unconfigured
- Condition: unknown

Verify this by using the `cEgadm` command to list the slots, as in the following example. In the dom1 domain, slot SB2 is to contain the CPU board:

- 3** Use the `cfgadm` command to connect and configure a CPU or memory board:

```
cfgadm -v -c configure SBx
```

For example:

```
# cfgadm -v -c configure SB2

assign SB2
assign SB2 done
poweron SB2
poweron SB2 done
test SB2
test SB2 done
connect SB2
connect SB2 done
configure SB2
configure SB2 done
notify online SUNW_cpu/cpu448
notify online SUNW_cpu/cpu449
notify online SUNW_cpu/cpu450
notify online SUNW_cpu/cpu451
notify add capacity (4 cpus)
notify add capacity (2097152 pages)
notify add capacity SB2 done
```

- 4** Verify the new board has been connected and configured using the command `cfgadm`. For example:

```
# cfgadm

Ap_Id Type Receptable Occupant Cond
.
SB2 CPU connected configured ok
```

Performing dynamic reconfiguration on PCI cards on I/O boards

A card containing an HBA can be removed and replaced on an I/O board. If a failed HBA has been used with other adapters on separate cards in a Dynamic Multi-Pathing (DMP) configuration, I/O can proceed through the alternate path and VCS need not be stopped.

To determine the status of the card you are unconfiguring

- 1 Log in to the domain as the administrator. For the following example, the I/O board is in the dom1 domain.
- 2 Check the status of the boards. Use the `cfgadm` command.

```
cougar# cfgadm
```

The output resembles:

```
Ap_Id Type Receptacle Occupant Condition
IO4 HPCI connected configured ok
IO4_C3V0 fibre/hp connected configured ok
IO4_C3V1 pci-pci/hp connected configured ok
IO4_C5V0 pci-pci/hp connected configured ok
IO4_C5V1 fibre/hp connected configured ok
SB7 CPU connected configured ok
SB8 CPU connected configured ok
c0 scsi-bus connected configured unknown
c1 scsi-bus connected unconfigured unknown
c2 fc connected unconfigured unknown
c3 fc connected unconfigured unknown
c4 fc-fabric connected configured unknown
c5 fc connected unconfigured unknown
cougar# uname -a
SunOS cougar 5.10 Generic_118833-17 sun4u sparc
SUNW,Sun-Fire-15000
cougar#
```

In case of Solaris, the reporting of I/O board slot names makes it somewhat easier to discover the relationship between physical and logical devices because slots on the I/O boards are also numbered using the C[35]V[01] notation.

To remove a PCI card

- 1 Disable the controllers on the I/O system card using the `vxdmpadm` command:

```
# vxdmpadm disable ctlr=c3
```

If the card has more than one controller, repeat this command for each controller on the card.

- 2 Disconnect the card:

```
# cfgadm -v -c disconnect pcisch1:sg8slot0
```

- 3 Check the states and the condition of the card using the `cfgadm` command:

```
# cfgadm
```

The disconnected card must have the following states and condition:

- Receptacle state: disconnected
- Occupant state: unconfigured
- Condition: unknown

4 Remove the disconnected card only if it is powered off.

To add a card

1 Verify that the slot you selected can accept a device, such as a PCI card.

To accept a device, the slot must have the following states and condition:

- Receptacle state: empty or disconnected
- Occupant state: unconfigured
- Condition: unknown

Verify this by using the `cfgadm` command to list all of the system boards, as in the following example:

The output resembles:

```
cougar# cfgadm

Ap_Id Type Receptacle Occupant Condition
IO4_HPCI connected configured ok
IO4_C3V0 fibre/hp connected configured ok
IO4_C3V1 pci-pci/hp connected configured ok
IO4_C5V0 pci-pci/hp connected configured ok
IO4_C5V1 fibre/hp connected configured ok
SB7 CPU connected configured ok
SB8 CPU connected configured ok
c0 scsi-bus connected configured unknown
c1 scsi-bus connected unconfigured unknown
c2 fc connected unconfigured unknown
c3 fc connected unconfigured unknown
c4 fc-fabric connected configured unknown
c5 fc connected unconfigured unknown
cougar# uname -a
SunOS cougar 5.10 Generic_118833-17 sun4u sparc
SUNW,Sun-Fire-15000
cougar#
```


In case of Sol 10, the reporting of I/O board slot names makes it somewhat easier to discover the relationship between physical and logical devices because slots on the I/O boards are also numbered using the C[35]V[01] notation.

- 2 Add the replacement PCI card to the empty card slot.
- 3 To configure the new card, use the `cfgadm` command. For example:

For s6800:

```
# cfgadm -c configure pcisch1:sg8slot0
```

For e12K/15K:

```
# cfgadm -c configure pcisch1:e15b1slot0
```

After the system configures and tests the board, it displays a message in the domain console log indicating the configuration of the components.

- 4 Check the states and the condition of the board using the `cfgadm` command; it must be “connected,” “configured,” and “ok.”
- 5 Enable the controller for the HBA:

```
# vxdmpadm enable ctlr=c3
```

Note: This command succeeds if the controller is accessible to the domain and I/O can be performed on it.

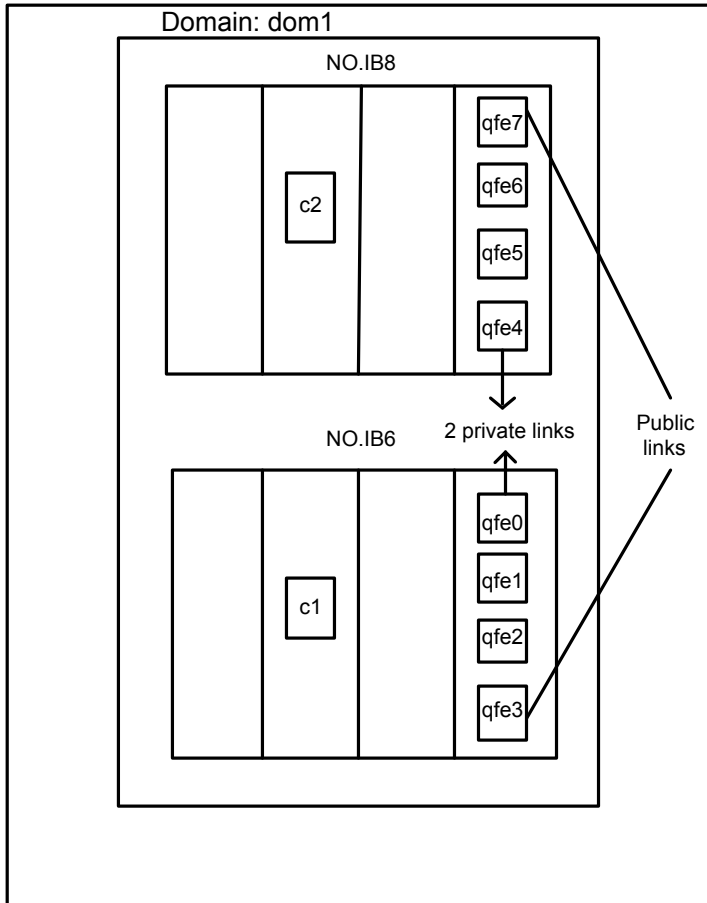
Performing dynamic reconfiguration on I/O boards

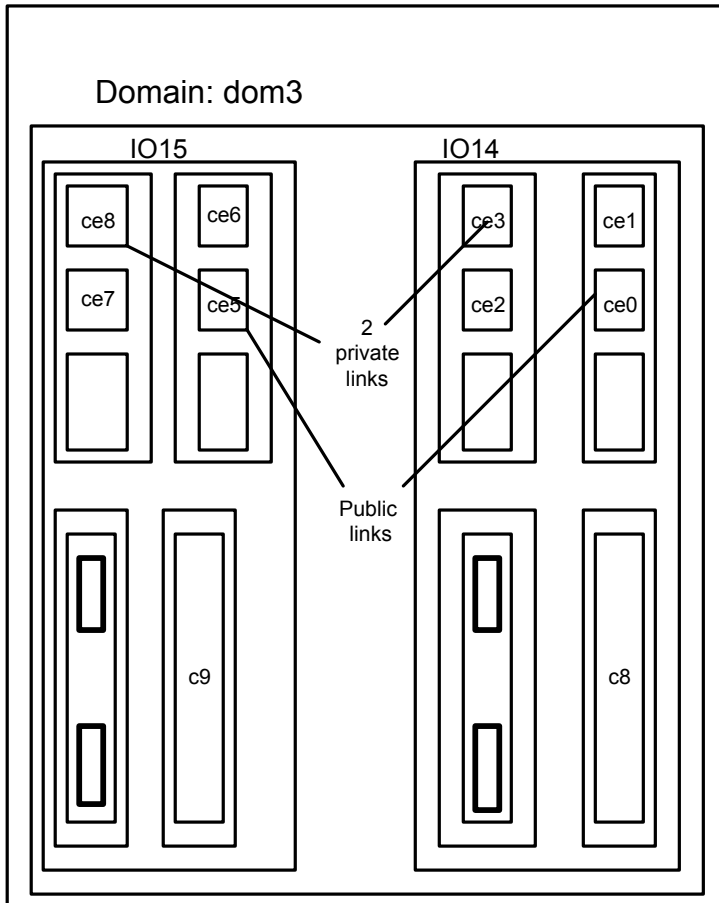
Under certain circumstances, you must stop VCS on the domain where you are reconfiguring a board.

See [“Scenarios requiring a VCS shutdown”](#) on page 6.

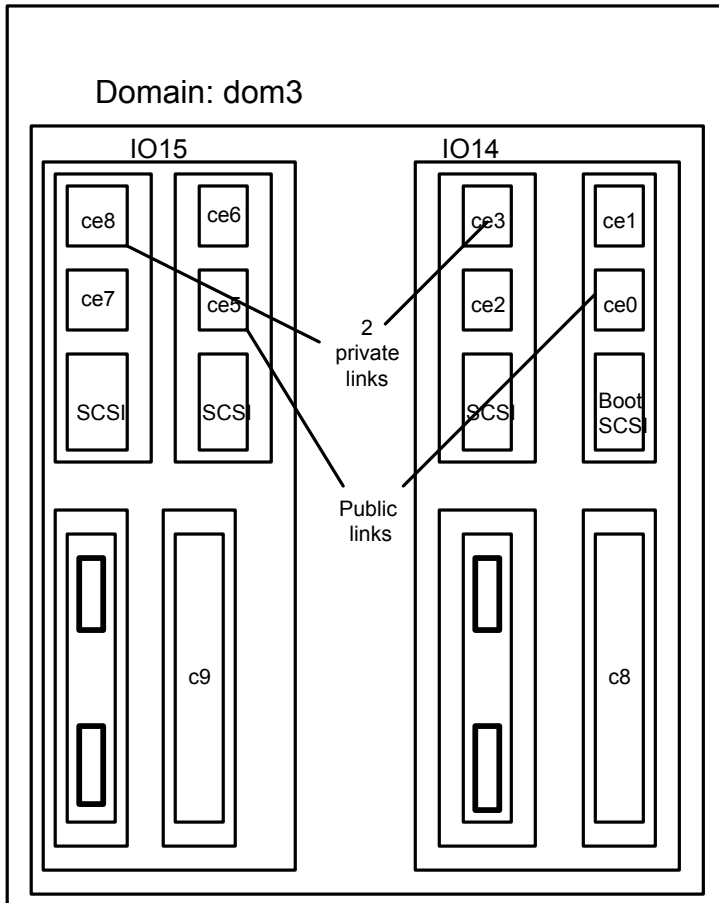
For s6800:

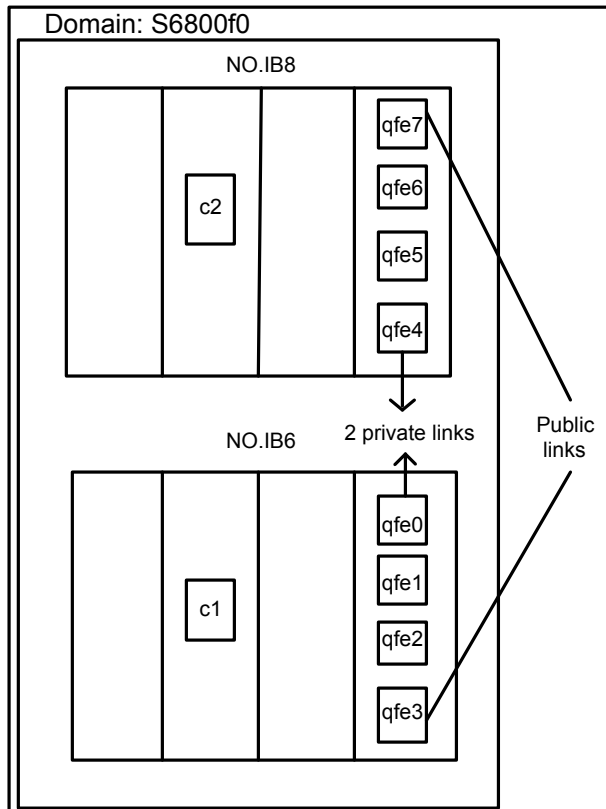
In the following scenario, a cluster consists of the dom1 and the dom3 domains. The cluster is running service groups on the dom1 domain, which includes I/O boards N0.IB8 and N0.IB6. N0.IB8 requires dynamic reconfiguration because of a malfunctioning component. The domain dom3 includes I/O boards IO14 and IO15. The disk controllers and NICs are labeled in the following diagrams.





For e12K/15K/25K: In the following scenario, a cluster consists of the dom3 and the S6800f0 domains. The cluster is running service groups on the dom3 domain, which includes I/O boards IO14 and IO15. IO15 requires dynamic reconfiguration because of a malfunctioning component. The domain S6800f0 includes I/O boards IB8 and IB6. The disk controllers and NICs are labeled in the following diagrams.





The highlights of the procedure to dynamically reconfigure the I/O boards (NO.IB8 board and IO15 board) in the dom1 and dom3 domains for s6800 and e12K/15K/25K respectively include:

- Disabling all the active controllers on the board.
- Disabling all the NIC devices used for private communications on the board
- Disabling all the NIC devices used for public communications on the board
- Disabling the IO board and removing it
- Adding the replacement IO board
- Enabling the replacement board
- Enabling the public NIC devices
- Enabling the private NIC devices
- Enabling the active controllers

To verify the status of the cluster before dynamic reconfiguration

- 1** Use the VCS command `hastatus -sum` to verify the current state of the service groups in the cluster. Use the command before reconfiguring the I/O board and after reconfiguration to verify the cluster's state. The output is as follows with slight variations for the different Oracle servers.

```
-- SYSTEM STATE
-- System State Frozen
A dom3 RUNNING 0
A s6800f0 RUNNING 0
-- GROUP STATE
-- Group System Probed AutoDisabled State
B ServiceGroupA dom3 Y N ONLINE
B ServiceGroupA s6800f0 Y N OFFLINE
B cvm dom3 Y N ONLINE
B cvm s6800f0 Y N ONLINE
```

- 2** For s6800: By using the `cfgadm -lv` command, you can show the I/O boards and cards in the dom1 domain. For example:

```
# cfgadm -lv
```

In the output (not shown), the board N0.IB8 is reported to be connected, configured, and ok. In addition, the condition of each of the slots on N0.IB8 are reported.

For e12K/15K: By using the `cfgadm -al` command, you can show the I/O boards and cards in the dom3 domain. For example:

```
# cfgadm -al
```

```
Ap_Id Type Receptacle Occupant
Condition
IO14 HPCI connected configured ok
IO14::pci0 io connected configured ok
IO14::pci1 io connected configured ok
IO14::pci2 io connected configured ok
IO14::pci3 io connected configured ok
IO15 HPCI connected configured ok
IO15::pci0 io connected configured ok
IO15::pci1 io connected configured ok
IO15::pci2 io connected configured ok
IO15::pci3 io connected configured ok
SB14 CPU connected configured ok
SB14::cpu0 cpu connected configured ok
.
.
.
pcisch1:e14b1slot0 fibre/hp connected configured ok
pcisch2:e14b1slot3 pci-pci/hp connected configured ok
pcisch3:e14b1slot2 ethernet/hp connected configured ok
pcisch4:e15b1slot1 pci-pci/hp connected configured ok
pcisch5:e15b1slot0 fibre/hp connected configured ok
pcisch6:e15b1slot3 pci-pci/hp connected configured ok
pcisch7:e15b1slot2 ethernet/hp connected configured ok
```

To determine the controllers on a board

- 1 Use the command `vxdmpadm listctlrall` to determine all controllers in the domain. For example, on the dom3 domain:

```
# vxdmpadm listctlr all

CTLR-NAME ENCLR-TYPE STATE ENCLR-NAME
=====
c0 Disk ENABLED Disk
c9 HDS9960 ENABLED HDS99600
c8 HDS9960 ENABLED HDS99600
```


- 2** To determine which controllers are on a specific board, for example IO15, use the following commands to display information about the disks in the domain, their controllers, and the location of the controllers on the IO boards.

Use the command `cfgadm -lv`, which provides a verbose listing of all boards in the domain. In the output, you can see the device slots listed for the board IO15.

```
# cfgadm -lv
```

In the following example (not all output is shown) the listing might contain lines that resemble:

```
.
pcish4:e15b1slot1 . . .
/devices/pci@1fc,700000:e15b1slot1
pcish5:e15b1slot0 . . .
/devices/pci@1fc,600000:e15b1slot0
pcish6:e15b1slot3 . . .
/devices/pci@1fd,700000:e15b1slot3
pcish7:e15b1slot2 . . .
/devices/pci@1fd,600000:e15b1slot2
.
```

The listing indicates that the device labeled `pci@1fc` is used by slots 0 and 1 of board 15, the device labeled `pci@1fd` is used by slots 3 and 2.

Using the `format` command in the domain, you can list the disk devices. The listing may be lengthy, but in the output, the controller, indicated by "c#" in the first two characters of the device name, corresponds to a device that is listed in the previous command (step a). For example:

```
# format

c0t0d0 <SUN18G ..... /pci@1dc,700000/pci@1.. .....
c8t0d0 <HITACHI-OPEN ....
/pci@1dc,600000/fibre-channel ...
.
c9t0d0 <HITACHI-OPEN ....
/pci@1fc,600000/fibre-channel ...
```

A comparison of the output of the previous two commands shows that board 15 slot 0 contains the controller c9.

- 3** As an alternative to using the `format` command, you can also use the following procedure to determine which storage controllers are impacted by dynamic reconfiguration on a given slot or I/O board for e25K on Solaris.

Verify which I/O controllers are impacted by dynamic reconfiguration on the board IO4 on sol10 (cougar) by using the following command:

```
cougar# cfdgadm -s "cols=ap_id:physid" | grep IO4

IO4 /devices/pseudo/dr@0:IO4
IO4_C3V0 /devices/pci@9c,600000:IO4_C3V0
IO4_C3V1 /devices/pci@9d,600000:IO4_C3V1
IO4_C5V0 /devices/pci@9c,700000:IO4_C5V0
IO4_C5V1 /devices/pci@9d,700000:IO4_C5V1
```

The `-s` parameter is used to limit output to the `ap_id` and physical id columns.

Notice the `pci@...` In the phys id, use `grep` again using `pci@9[cd],[67]00000`:

```
cougar# cfdgadm -s "cols=ap_id:physid" | grep pci@9[cd],[67]

IO4_C3V0 /devices/pci@9c,600000:IO4_C3V0
IO4_C3V1 /devices/pci@9d,600000:IO4_C3V1
IO4_C5V0 /devices/pci@9c,700000:IO4_C5V0
IO4_C5V1 /devices/pci@9d,700000:IO4_C5V1
c0 /devices/pci@9c,700000/pci@1/scsi@2:scsi
c1 /devices/pci@9c,700000/pci@1/scsi@2,1:scsi
c2 /devices/pci@9c,600000/SUNW,qlc@1,1/fp@0,0:fc
c3 /devices/pci@9c,600000/SUNW,qlc@1/fp@0,0:fc
c4 /devices/pci@9d,700000/SUNW,qlc@1/fp@0,0:fc
c5 /devices/pci@9d,700000/SUNW,qlc@1,1/fp@0,0:fc
```

`c0` and `c1` are located on `IO4_C5V0`, `c2` and `c3` are on `IO4_C3V0`, and `c4` and `c5` are on `IO4_C5V1`

On sol 9, the procedure is almost the same:

```
jaguar# cfdgadm -s "cols=ap_id:physid" | grep e17
```

e17 corresponds to the IO board #17

```
pcisch4:e17b1slot1 /devices/pci@23c,700000:e17b1slot1
pcisch5:e17b1slot0 /devices/pci@23c,600000:e17b1slot0
pcisch6:e17b1slot3 /devices/pci@23d,700000:e17b1slot3
pcisch7:e17b1slot2 /devices/pci@23d,600000:e17b1slot2

jaguar# cfdgadm -s "cols=ap_id:physid" | grep pci@23[cd],[67]
```

```
c4 /devices/pci@23c,700000/pci@1/scsi@2:scsi
c5 /devices/pci@23c,700000/pci@1/scsi@2,1:scsi
c6 /devices/pci@23d,700000/SUNW,qlc@1/fp@0,0:fc
pcisch4:e17b1slot1 /devices/pci@23c,700000:e17b1slot1
pcisch5:e17b1slot0 /devices/pci@23c,600000:e17b1slot0
pcisch6:e17b1slot3 /devices/pci@23d,700000:e17b1slot3
pcisch7:e17b1slot2 /devices/pci@23d,600000:e17b1slot2
```

c4 and c5 are on e17b1slot1 and c6 is on slot3

To determine the network interfaces on the board

- ◆ Verify which network interfaces correspond to which slot on the I/O board (since each I/O board can carry upto four PCI cards) by using the `grep` command to match the `/etc/path_to_inst` for pci identifiers.

For e25K on Solaris

```
IO4_C3V0 /devices/pci@9c,600000:IO4_C3V0
IO4_C3V1 /devices/pci@9d,600000:IO4_C3V1
IO4_C5V0 /devices/pci@9c,700000:IO4_C5V0
IO4_C5V1 /devices/pci@9d,700000:IO4_C5V1
```

```
cougar# grep pci@9[cd],[67] /etc/path_to_inst |grep network
```

```
"/pci@9c,700000/network@3,1" 0 "eri"
"/pci@9c,700000/pci@1/network@0" 0 "ce"
"/pci@9c,700000/pci@1/network@1" 1 "ce"
"/pci@9d,600000/pci@1/network@0" 2 "ce"
```

IO4_C5V0 contains eri0, c0, and c1. IO4_C3V1 contains ce2.

```
cougar#
```

To disable the controllers on the board

- 1** Disable the active controllers on the I/O system card using the `vxdmpadm` command.

```
vxdmpadm disable ctlr=ctlr
```

For s6800:

```
# vxdmpadm disable ctlr=c2
```

For e12K/15K:

```
# vxdmpadm disable ctlr=c9
```

- 2** Using the `vxdmpadm` command, verify that the controller is disabled. The output for all Oracle servers (s6800 and e12K/15K/25K) will be similar except for minor differences.

```
# vxdmpadm listctlr all
```

For s6800: In this example, the only controller on board is c2.

```
CTLR-NAME ENCLR-TYPE STATE ENCLR-NAME
=====
c0 Disk ENABLED Disk
c2 HDS9960 DISABLED HDS99600
c1 HDS9960 ENABLED HDS99600
```

For e12K/15K: In this example, the only controller on board IO15 is c9.

```
CTLR-NAME ENCLR-TYPE STATE ENCLR-NAME
=====
c0 Disk ENABLED Disk
c9 HDS9960 DISABLED HDS99600
c8 HDS9960 ENABLED HDS99600
```

- 3** If a card has more than one controller, repeat this command for each controller on the card to be reconfigured.

To list the status of the private network links and to disable them

- 1 Enter the command `lltstat -nv`:

The output resembles:

For s6800:

```
LLT node information:
Node State Links
* 0 dom1 OPEN 2
1 dom3 OPEN 2
2 CONNWAIT 0
.
.
31 CONNWAIT 0
```

The output shows that both domains have two links for private communication. Both links are “OPEN,” that is, operational.

For e12K/15K:

```
LLT node information:
Node State Links
0 s6800f0 OPEN 2
* 1 dom3 OPEN 2
2 CONNWAIT 0
.
.
31 CONNWAIT 0
```

The output shows that both domains have two links for private communication. Both links are “OPEN,” that is, operational.

- 2 Display the /etc/llttab file using the following command:

```
# cat /etc/llttab
```

For s6800:

```
set-node dom1  
set-cluster 13  
link qfe4 /dev/qfe:4 - ether - -  
link qfe0 /dev/qfe:0 - ether - -
```

The devices qfe0 and qfe4 are shown as the private network links.

For e12K/15K:

```
set-node dom3  
set-cluster 13  
link cd3 /dev/ce:3 - ether - -  
link cd8 /dev/ce:8 - ether - -
```

The devices ce3 and ce8 are shown as the private network links.

3 Disable the private network link device.

For example for s6800, the private network link device is: qfe4, on I/O board N0.IB8.

```
# /sbin/lltconfig -u qfe4
```

For example for e12K/15K, the private network link device is: ce8, on I/O board 15.

```
# /sbin/lltconfig -u ce8
```

4 Check the status of the private network links:

```
# lltstat -nv
```

For s6800:

```
LLT node information:
Node State Links
* 0 dom1 OPEN 2
dom3 OPEN 1
2 CONNWAIT 0
.
.
.
31 CONNWAIT 0
```

For e12K/15K:

```
LLT node information:
Node State Links
0 s6800f0 OPEN 1
* 1 dom3 OPEN 2
2 CONNWAIT 0
.
.
.
31 CONNWAIT 0
```

To list the status of the public NICs and to disable them

- 1 Use the command `ifconfig -a`.

For s6800: For example, `qfe3` (on board N0.IB6) and `qfe7` (on board N0.IB8), the NICs used for the public network connections, are operational.

```
# ifconfig -a

lo0: flags=1000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4> mtu 8232
index
1 inet 127.0.0.1 netmask ff000000
ge0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500
index 2 inet 10.182.65.99 netmask fffff000 broadcast
10.182.79.255 ether 0:3:ba:8:ec:40
qfe3:
flags=9040843<UP,BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4,
NOFAILOVER> mtu 1500 index 3 inet 10.182.66.143 netmask
ffffff00 broadcast 10.255.255.255 groupname mn1 ether
0:3:ba:8:ec:40
qfe7:
flags=9040843<UP,BROADCAST,RUNNING,MULTICAST,DEPRECATED,IPv4,
NOFAILOVER> mtu 1500 index 4 inet 10.182.66.144 netmask
ffffff00 broadcast 10.255.255.255 groupname mn1 ether
0:3:ba:8:ec:40
```

- 2 For s6800: To disable the device `qfe7` on board N0.IB8, use the commands:

```
# ifconfig qfe7 down
# ifconfig qfe7 unplumb
```

For e12K/15K: To disable the device `ce5` on board IO15, use the command:

```
# ifconfig ce5 down
```

- 3 For s6800: Use the `ifconfig -a` command to verify that `qfe7` is down. No information about `qfe7` should appear in the output.

For e12K/15K/25K: Use the `ifconfig -a` command to verify that `ce5` is down. No information about `ce5` should appear in the output.

```
# ifconfig -a
```


To disable and remove the IO board

- 1** When the controllers and network interface cards are disabled, disconnect the board:

For s6800:

```
# cfgadm -c disconnect N0.IB8
```

For e12K/15K:

```
# cfgadm -c disconnect IO15
```

Note: The -f option is recommended only when a normal disconnect attempt fails and there is no clear way to make the command succeed gracefully.

- 2** Use the `cfgadm` command to check the status of the I/O board:

```
# cfgadm -al
```

For s6800: In the output, the fields Receptacle, Occupant, and Condition for N0.IB8 show disconnected, unconfigured, and unknown respectively.

The I/O board may be physically removed at this time. Before adding the new board to the dom1 domain, you must test it in another spare domain.

For e12K/15K:

```
Ap_Id Type Receptacle Occupant
Condition
IO14 HPCI connected configured ok
IO14::pci0 io connected configured ok
IO14::pci1 io connected configured ok
IO14::pci2 io connected configured ok
IO14::pci3 io connected configured ok
IO15 HPCI disconnected unconfigured
unknown
SB14 CPU connected configured ok
SB14::cpu0 cpu connected configured ok
.
.
```

The I/O board, IO15, may be physically removed at this time.

To add the new IO board

- 1 Physically add the board, connecting all necessary cables, and configure it:

For s6800:

```
# cfgadm -c configure N0.IB8
```

For e12K/15K:

```
# cfgadm -c configure IO15
```

Note: Make sure that the output of the `cfgadm` command shows the slot where the new board is to be added. The status is disconnected, unconfigured, and unknown.

- 2 Run the `cfgadm -al` command to verify the board has been configured; the board should be connected, configured, and ok. If you have stopped VCS, you may skip step 3 through step 6.

- 3 Reconfigure the network interface cards on the new board:

For s6800:

```
# ifconfig qfe7 plumb
```

```
# ifconfig qfe7 up
```

For e12K/15K:

```
# ifconfig ce5 plumb
```

- 4 Run the command `ifconfig -a` to verify that the NICs are up and running.

- 5 Reconfigure LLT to reestablish the private network links:

For s6800:

```
# /sbin/lltconfig -t qfe4 -d /dev/qfe:4
```

For e12K/15K:

```
# /sbin/lltconfig -t ce8 -d /dev/ce:8
```

- 6 Verify the private network links are restored using the command `lltstat -nv`:

```
# /sbin/lltstat -nv
```

7 For s6800: Enable the controller c2 on the N0.IB8 using `vxdmpadm` command:

```
# vxdmpadm enable ctrl=c2
```

For e12K/15K: Enable the controller c9 on the IO15 using `vxdmpadm` command:

```
# vxdmpadm enable ctrl=c9
```

8 Verify that the controller is up and running:

```
# vxdmpadm listctrl all
```

If you have stopped VCS before reconfiguring the I/O board, restart it. Refer to the section, See [“Stopping and starting VCS”](#) on page 7.

Replacing an online Host Bus Adapter (HBA) on an M5000 server

This section contains the procedure to replace an online Host Bus Adapter (HBA) when DMP is managing multi-pathing in a Cluster File System (CFS) cluster. The HBA World Wide Port Name (WWPN) changes when the HBA is replaced.

Following are the prerequisites to replace an online Host Bus Adapter (HBA):

- A single node or two or more node CFS or RAC cluster.
- I/O running on CFS file system.
- An M5000 server with atleast two HBAs in separate PCIe slots and recommended Solaris patch level for HBA replacement.

Following is the procedure to hotswap an online Host Bus Adapter on an M5000 server:

To replace an online Host Bus Adapter (HBA) on an M5000 server

- 1 Identify the HBAs on the M5000 server using the following command:

```
/usr/platform/sun4u/sbin/prtdiag -v | grep emlx ( emulex HBA)
```

```
/usr/platform/sun4u/sbin/prtdiag -v | grep qlc ( qllogic HBA )
```

```
00 PCIe 0      2, fc20, 10df      119, 0, 0 okay      4,
4  SUNW,emlxs-pci10df,fc20      LPe 11002-S
   /pci@0,600000/pci@0/pci@9/SUNW,emlxs@0
```

```
00 PCIe 0      2, fc20, 10df      119, 0, 1 okay      4,
4  SUNW,emlxs-pci10df,fc20      LPe 11002-S
   /pci@0,600000/pci@0/pci@9/SUNW,emlxs@0,1
```

```
00 PCIe 3      2, fc20, 10df           2, 0, 0 okay      4,
4  SUNW,emlxs-pci10df,fc20      LPe 11002-S
   /pci@3,700000/SUNW,emlxs@0
```

```
00 PCIe 3      2, fc20, 10df           2, 0, 1 okay      4,
4  SUNW,emlxs-pci10df,fc20      LPe 11002-S
   /pci@3,700000/SUNW,emlxs@0,1
```

- 2** Identify the HBA and its WWPN(s), which you want to replace using the `cfgadm` command.

To identify the HBA:

```
# cfgadm -al | grep -i fibre
```

```
iou#0-pci#1 fibre/hp connected configured ok
```

```
iou#0-pci#4 fibre/hp connected configured ok
```

To list all HBAs:

```
# luxadm -e port ( will list all HBA's )
```

```
/devices/pci@0,600000/pci@0/pci@9/SUNW,emlxs@0/fp@0,0:devctl  
NOT CONNECTED
```

```
/devices/pci@0,600000/pci@0/pci@9/SUNW,emlxs@0,1/fp@0,0:devctl  
CONNECTED
```

```
/devices/pci@3,700000/SUNW,emlxs@0/fp@0,0:devctl  
NOT CONNECTED
```

```
/devices/pci@3,700000/SUNW,emlxs@0,1/fp@0,0:devctl  
CONNECTED
```

Select the HBA to dump the portap and get the WWPN:

```
# luxadm -e dump_map /devices/pci@0,600000/pci@0/pci@9/SUNW,emlxs@0,1/  
fp@0,0:devctl
```

```
0 304700 0 203600a0b847900c 200600a0b847900c 0x0  
(Disk device)
```

```
1 30a800 0 20220002ac00065f 2ff70002ac00065f 0x0  
(Disk device)
```

```
2 30a900 0 21220002ac00065f 2ff70002ac00065f 0x0  
(Disk device)
```

```
3 560500 0 10000000c97c3c2f 20000000c97c3c2f 0x1f  
(Unknown Type)
```

```
4 560700 0 10000000c97c9557 20000000c97c9557 0x1f  
(Unknown Type)
```

```
5 560b00 0 10000000c97c34b5 20000000c97c34b5 0x1f  
(Unknown Type)
```

```
6 560900 0 10000000c973149f 20000000c973149f 0x1f  
(Unknown Type, Host Bus Adapter)
```

Alternately, you can run the `fcinfo hba-port` Solaris command to get the WWPN(s) for the HBA ports.

- 3 Ensure you have a compatible spare HBA for hot-swap.
- 4 Stop the I/O operations on the HBA port(s) and disable the DMP subpath(s) for the HBA that you want to replace.

```
# vxddmpadm disable ctrl=<ctrl#>
```

- 5 Dynamically unconfigure the HBA in the PCIe slot using the `cfgadm` command.

```
# cfgadm -c unconfigure iou#0-pci#1
```

Look for console messages to check if the `cfgadm` command is unsuccessful.

If the `cfgadm` command is unsuccessful, proceed to troubleshooting using the server hardware documentation. Check the Solaris patch level recommended for dynamic reconfiguration operations and contact Oracle support for further assistance.

console messages

```
Oct 24 16:21:44 m5000sb0 pcihp: NOTICE: pcihp (pxb_plx2):  
card is removed from the slot iou 0-pci 1
```

- 6 Verify that the HBA card that is being replaced in step 5 is not in the configuration using the following command:

```
# cfgadm -al | grep -i fibre
```

```
iou 0-pci 4 fibre/hp connected configured ok
```

- 7 Mark the fiber cable(s).
- 8 Remove the fiber cable(s) and the HBA that you must replace.

Note: You can refer to the HBA replacement procedures in *SPARC Enterprise M4000/M5000/M8000/M9000 Servers Dynamic Reconfiguration (DR) User's Guide* for more information.

- 9** Replace it with a new compatible HBA of similar type in the same slot.

The reinserted card shows up as follows:

```
console messages
```

```
iou 0-pci 1 unknown disconnected unconfigured unknown
```

- 10** Run the following command to bring the replaced HBA back into the configuration.

```
# cfgadm -c configure iou 0-pci 1
```

```
console messages
```

```
Oct 24 16:21:57 m5000sb0 pcihp: NOTICE: pcihp (pxb_plx2):  
card is inserted in the slot iou#0-pci#1 (pci dev 0)
```

- 11** Verify that the reinserted HBA is in the configuration using the `cfgadm` command:

```
# cfgadm -al | grep -i fibre
```

```
iou#0-pci 1 fibre/hp connected configured ok <====
```

```
iou#0-pci 4 fibre/hp connected configured ok
```

- 12** Modify fabric zoning to include the replaced HBA WWPN(s).

- 13** Enable LUN security on storage for the new WWPN(s).

- 14** Perform an operating system device scan to re-discover the LUNs using the `cfgadm` command:

```
# cfgadm -c configure c3
```

- 15** Clean up the device tree for old LUNs.

```
# devfsadm -Cv
```

Note: Sometimes HBA replacement may create new devices. Perform cleanup operations for the LUN only when new devices are created.

- 16** If VxVM / Dynamic Multi-pathing (DMP) does not show a ghost path for the removed HBA path, enable the path using the `vxdmpadm` command: This performs the device scan for that particular HBA subpath(s).

```
# vxdmpadm disable ctrl=<ctrl#>
```

- 17** Verify if I/O operations are scheduled on that path.
If I/O operations are running correctly on all paths, then the dynamic HBA replacement operation is complete.