Dynamic Reconfiguration
User's Guide for Sun™
Enterprise™ 6x00, 5x00, 4x00,
and 3x00 Systems

**Solaris™ Version 2.6**
**Hardware: 5/98**

Send comments about this document to: smcc-docs@sun.com

Adobe PostScript™

# Contents

# Preface

This document describes the Dynamic Reconfiguration (DR) feature, which enables you to logically attach and detach system boards from a Sun™ Enterprise™ 3*x*00⁄ 4*x*00⁄5*x*00⁄6*x*00 server while the server continues running.

## Sun Documentation on the Web

For the latest information about supported hardware, firmware, known bugs, and documentation errata for dynamic reconfiguration, refer to the Current Release (Version #) in the web site:

```
http://sunsolve2.Sun/COM/beta/DR/
```

The `docs.sun.com` web site enables you to access Sun technical documentation on the World Wide Web. You can browse the `docs.sun.com` archive or search for a specific book title or subject at `http://docs.sun.com`.

# Using UNIX Commands

This document may not contain information on basic UNIX® commands and procedures such as shutting down the system, booting the system, and configuring devices.

See one or more of the following for this information:

- *Solaris 2.x Handbook for SMCC Peripherals* (*If you are incorporating Solaris 2.x software commands in your document, delete this sentence.*)
- AnswerBook™ online documentation for the Solaris™ 2.x software environment
- Other software documentation that you received with your system

# Typographic Conventions

**Table P-1**

| Typeface or Symbol | Meaning | Examples |
|---|---|---|
| `AaBbCc123` | The names of commands, files, and directories; on-screen computer output. | Edit your `.login` file.<br>Use `ls -a` to list all files.`% You have mail.` |
| **`AaBbCc123`** | What you type, when contrasted with on-screen computer output. | `%` **`su`**`Password:` |
| *AaBbCc123* | Book titles, new words or terms, words to be emphasized. Command-line variable; replace with a real name or value. | Read Chapter 6 in the *User's Guide*.<br>These are called *class* options.<br>You *must* be `root` to do this.<br>To delete a file, type `rm` *filename*. |

# Shell Prompts

**Table P-2**

| Shell | Prompt |
| --- | --- |
| C shell | *machine_name*% |
| C shell superuser | *machine_name*# |
| Bourne shell and Korn shell | $ |
| Bourne shell and Korn shell superuser | # |

# Ordering Sun Documents

SunDocsSM is a distribution program for Sun Microsystems technical documentation. Contact SunExpress for easy ordering and quick delivery. You can find a listing of available Sun documentation on the World Wide Web.

**Table 0-1**

| Country | Telephone | Fax |
| --- | --- | --- |
| Belgium | 02-720-09-09 | 02-725-88-50 |
| Canada | 1-800-873-7869 | 1-800-944-0661 |
| France | 0800-90-61-57 | 0800-90-61-58 |
| Germany | 01-30-81-61-91 | 01-30-81-61-92 |
| Holland | 06-022-34-45 | 06-022-34-46 |
| Japan | 0120-33-9096 | 0120-33-9097 |
| Luxembourg | 32-2-720-09-09 | 32-2-725-88-50 |
| Sweden | 020-79-57-26 | 020-79-57-27 |
| Switzerland | 0800-55-19-26 | 0800-55-19-27 |
| United Kingdom | 0800-89-88-88 | 0800-89-88-87 |
| United States | 1-800-873-7869 | 1-800-944-0661 |

World Wide Web: `http://www.sun.com/sunexpress/`

# Sun Welcomes Your Comments

You can email or fax your comments to us. Please include the part number of your document in the subject line of your email or fax message.

- Email:smcc-docs@sun.com
- Fax:SMCC Document Feedback
  1-650-786-6443

# Introduction to Dynamic Reconfiguration

This chapter briefly describes Dynamic Reconfiguration (DR). The features and commands described here are specific to Sun™ Enterprise™ 3*x*00, 4*x*00, 5*x*00, and 6*x*00 servers and may not apply to other types of servers.

**Note –** This release of DR works with hot-swap I/O boards. This release of DR does not support CPU/Memory boards.

## Overview

DR software works with "hot swap" hardware. You can reconfigure hot-swappable boards in the server without needing to reboot. You can remove or replace I/O boards while applications continue to run on other boards within the server.

The release of Dynamic Reconfiguration (DR) software allows you to:

- Display the status of I/O boards and adapters
- Test I/O boards and adapters
- Add an I/O board or adapter
- Remove or replace a faulty I/O board or adapter
- Isolate failed I/O hardware before it can cause problems

# What Is Dynamic Reconfiguration?

Dynamic Reconfiguration is a set of enhancements to Solaris software that enable hardware resource alterations on a server while it continues to run. This release of DR enables two capabilities on Sun Enterprise *x*000 servers:

- To add or upgrade a system board
- To remove a faulty system board

DR requires manual intervention from an administrator. It cannot reconfigure components automatically.

For example, when you insert a new PCI I/O board into a backplane slot:

1. Look for DR to display a console message such as, "`PCI board inserted into slot3.`"

2. Configure the new board by using the DR configuration administration interface to issue a command such as:

```
# cfgadm -c configure sysctrlx:slotx
```

3. Reconfigure the devices on the newly-inserted board using standard Solaris commands such as `drvconfig` and `devlinks`.

4. Activate these devices using standard Solaris commands such as `mount` and `ifconfig`.

Note that a reboot is not required and applications can continue to run during this process (provided there are enough hardware resources in the system.)

DR also allows the replacement of failed hardware, and reactivation of the replacement hardware after repair. If a replacement is not immediately available, you can isolate failed hardware before it can cause system problems.

Similarly, DR allows the removal of an I/O board without requiring an operating system reboot. The key to system board removal, as explained in Chapter 2, is making sure that devices attached to a system board are inactive.

# Logical Versus Physical DR

DR can also allow the dynamic reconfiguration of hardware resources without actual physical hardware additions or removals. **Physical DR** involves the insertion or removal of a board. **Logical DR** activates or deactivates a board, and does not necessarily require a hardware change.

# DR and Attachment Points

Dynamic Reconfiguration actually occurs at any point within a system where hardware resources can be added to or subtracted from the machine. These points are called *attachment points*. An attachment point can be a physical or a logical locale. Software support for DR must exist at a given point in order for that point to be an attachment point. Furthermore, hardware and software hot plug support must exist for that physical attachment point.

## The Components of Attachment Points

A DR attachment point consists of two parts:

- **receptacle**—a receiver, usually a board slot.
- **occupant**—a removable system board and any additional components which either reside on the board or are connected to the board. This includes any storage devices connected by cable to the board.

Receptacles can be designed to accept a single occupant or multiple occupants. Receptacles can have unique names, typically geographic locations like slot numbers. Or, receptacles can be anonymous like a traditional SCSI chain. Receptacles may have the ability to electrically isolate the occupant from the host machine. Occupants may contain more than one device.

There is one attachment point per board slot. An attachment point is named after the slot which it controls.

# The States and Conditions of DR Attachment Points

The configuration administration interface permits the alteration of the state of any attachment point component (receptacle or occupant), as well as viewing the condition of a complete attachment point (receptacle *and* occupant).

TABLE 1-1 describes the allowable states of receptacles. The transition model for receptacle states is linear and must be used in the order shown in the table.

**TABLE 1-1**  Receptacle States

| State | Description |
|---|---|
| empty | No board is present in the slot. All LEDs are off. |
| disconnected | A board is inserted but electrically disconnected. The system can identify the presence of a board and its board type. The LEDs show that the board is in low power mode and can be unplugged at any time.<br>LED:   green   yellow  green<br>         Off      On      Off<br>Use `cfgadm -c disconnect` to enable this state. |
| connected | A board is inserted and electrically connected and powered up. The system is actively monitoring the board for temperature and cooling.<br>LED:   green   yellow  green<br>         On      Off      Off<br>Use `cfgadm -c connect` to enable this state. |

TABLE 1-2 lists the possible states of occupants.

**TABLE 1-2**  Occupant States

| State | Description |
|---|---|
| configured | Devices are known to the system and have been fully initialized and may be mounted or configured for use.  The LEDs show the normal OS running pattern.<br>LED:   green   yellow  green<br>         On      Off      Blink<br>Use `cfgadm -c configure` to enable this state. |
| unconfigured | All other situations of device state including devices with their receptacles in the Empty state (i.e. devices which are physically not present)  LED pattern is the same for Connected/Unconfigured<br>LED:   green   yellow  green<br>         On      Off      Off<br>Use `cfgadm -c unconfigure` to enable this state. |

There are two types of attachment points, physical and logical.

A **physical attachment point** is represented by a minor device of type DDI_NT_ATTACHMENT_POINT. These attachment points reside as minor nodes to the system clock. An example of an attachment point device path is:

```
/devices/central@1f,0/fhc@0,f8800000/clock-board@0,900000:sysctrl,slot0
```

There is one attachment point per slot. Attachment points are named after the slot which it controls.

A **logical attachment point** is the abbreviated name of the physical attachment point mentioned above. The logical attachment point looks like this:

```
sysctrl0:slot0
```

To obtain a list of all available logical attachment points, use the command:

```
cfgadm -l
```

cfgadm recognizes both logical and physical attachment points.

TABLE 1-3 lists the five possible **attachment point conditions**:

**TABLE 1-3**    Attachment Point Conditions

| Condition | Description |
| --- | --- |
| Unknown | The current condition cannot be determined because nothing has ever been connected at the attachment point. This situation results either from inserting a new board after cold start, or a board that is placed on the disabled board list prior to cold start. A connected transition will change an attachment point's condition from Unknown to either OK or Failed. |
| OK | No ailing factors have been detected. This condition can only occur after a board has been connected. This condition will also persist until the board is either physically removed, or an ailing factor is discovered. Factors that are considered for Sunfire in order to be OK are: correct hardware compatibility, correct firmware revision, adequate power, adequate cooling, and adequate precharge. |

**TABLE 1-3** Attachment Point Conditions *(Continued)*

| Condition | Description |
|-----------|-------------|
| Failing | can only occur after a board has been connected and an ailing factor was discovered after being in the OK condition. This means that the board associated with the attachment point is over heating. This condition will persist until either the ailing factor disappears (no longer overheated) or the attachment point is disconnected. This condition is considered correctable. |
| Failed | May occur either from cold start or from a failed connect attempt. The board has failed POST/OBP. This condition will persist until the board is physically removed. This condition is considered uncorrectable and the attachment point should never transition beyond Disconnected. |
| Unusable | Denotes either that an attachment point has incompatible hardware, or that an empty attachment point either lacks power, cooling or precharge. Whenever a board enters the unusable condition, this situation is always correctable. Unusable condition is set whenever one of the following events occur: Inadequate cooling or power is detected for an empty slot; Inadequate cooling, Inadequate power, or unsupported hardware is present in a disconnected board; firmware detected a problem either during cold start or during hot insertion. |

# Connection and Configuration

For adding resources DR requires two key operations:

- Connection of a receptacle
- Configuration of the receptacle's occupant

The reverse process—removing resources—also requires two similar operations:

- Unconfiguration of a receptacle's occupant
- Disconnection of the receptacle

The DR configuration administration interface can perform both key operations with a single command. When necessary, this interface allows performing either operation separately.

## Connection Versus Configuration

To understand the difference between connection and configuration, think about a three-way lamp. Plugging the lamp into an electrical outlet is a physical attachment to the electrical power grid. But if the circuit breaker for the outlet is turned off, the outlet is like a DR attachment point receptacle that is not connected. Engaging the circuit breaker connects the outlet to power. This operation resembles connecting a DR attachment point receptacle.

The three-way lamp can now receive power. But what light level will it provide—low, medium, or high? An operator must configure the lamp by setting its activation switch to the proper position to produce the desired light level. This operation is similar to configuring a DR attachment point occupant. The power grid can now engage the device to produce a specified light level.

## The Purpose of Receptacle Connection/Disconnection

Connecting and disconnecting a DR attachment point receptacle makes that receptacle available or unavailable to a server's system bus. Boards can be plugged into a machine, but may not be accessible to the machine until the attachment point's receptacle is connected.

The DR configuration administration interface allows the performance of multiple state transitions with a single invocation. The `cfgadm -c configure` command can perform a connect on an unconnected receptacle as well as performing the configure operation. When used in this fashion, it may not be apparent that the connection operation occurred.

## The Purpose of Occupant Configuration/Unconfiguration

Configuring an attachment point occupant, usually a board, logically attaches that board to the operating system. A system board is logically attached when its resources—processors, memory, and I/O adapters—have been incorporated into the system's hardware configuration data structures. Attaching also loads device drivers for newly installed hardware. The system board must already be present in the system, and powered on. Normally, administrators attach a system board after it is inserted and powered on by an authorized Sun service provider.

Unconfiguring a board logically detaches that system board from the operating system. A system board is logically unconfigured when its resources—processors, memory, and I/O adapters—are removed from the system configuration and are no longer available. Unconfiguring also takes the device drivers offline for unconfigured hardware. Normally, administrators unconfigure a system board to prepare it for removal using hot swap by an authorized Sun service provider.

If a system board is being used by the operating system, an operator must stop its use and unconfigure it before it can be powered off and removed. After a new or upgraded system board is inserted and powered on, an operator can connect its attachment point and configure it for use by the operating system.

Operators can improve the chances of being able to disconnect the hardware resources connected to an I/O board by also using Alternate Pathing™ software.

# The Logical Steps for Using Dynamic Reconfiguration

The five logical operations required to use DR include:

- Determining the suitability of the target system boards
- Preparing attachment points for board insertion
- Inserting and removing boards or devices
- Connecting/disconnecting receptacles
- Configuring/unconfiguring attachment point occupants

The specific order for these logical operations is slightly different for adding a system board versus disabling or removing a system board. Determining the suitability of boards for using DR can be a complex subject and is covered in detail in Chapter 2.

# The Dynamic Reconfiguration User Interface

This release of DR uses a command-line user interface. The primary command for Dynamic Reconfiguration is the configuration administration interface and associated command, cfgadm (see cfgadm(1M)).

The cfgadm command performs operations on dynamically reconfigurable hardware resources. These operations include:

- Displaying status (-l)
- Initiating testing (-t)
- Invoking configuration state changes (-c)
- Invoking hardware specific functions that do not change state (-x)

For example, the cfgadm -c configure sysbrd0:slot4 command makes changes to the state of an attachment point. The value configure specifies in what state to place the attachment point identified by sysbrd0:slot4.

The structure of the configuration administration interface includes:

- The cfgadm command
- A generic library
- Separately supplied, dynamically loadable hardware-specific libraries

The cfgadm command and the generic library provide basic capabilities and stub routines for hardware-specific functions. Hardware-specific libraries provide routines called by DR operations in the generic library.

# A Dynamic Reconfiguration Scenario

After determining that a given hardware resource is usable with DR, the first task in a DR user scenario would be to ensure that the target receptacle is in a state that allows reception of an occupant. This state on a Sun Enterprise 3x00-6x00 server would be Empty. An administrator would use the cfgadm(1M) command to examine or alter the state of the receptacle.

Next, a service representative would insert the hardware resource. The receptacle changes to the Disconnected state. The occupant would be in the Unconfigured state. The cfgadm -c connect command can then change the receptacle to the Connected state. Finally, the cfgadm -c configure command can change the state of the occupant to Configured. At this point, the hardware resource is available for use by the Solaris environment through standard Solaris commands for configuring and using devices. No operating system reboot is necessary.

## Examples of Adding Hardware

There are three ways to add I/O hardware:

- **Insert a new I/O board**
- **Enable a Disabled I/O board**—activate an I/O board that was previously placed on the disabled board list
- **Add storage**—activate additional storage devices after a cold start

### Inserting a New I/O Board

Before an I/O board is inserted, the states and conditions are:

1. Receptacle state—Empty

2. Occupant state—Unconfigured

3. Condition—Unknown

After an I/O board is inserted, the states and conditions should become:

1. Receptacle state—Disconnected

2. Occupant state—Unconfigured

3. Condition—Unknown

Any other states or conditions should be considered an error.

In order to transition to the Connected state, the operator must execute:

```
# cfgadm -c connect logical ap_id
```

The states and conditions should be:

1. Receptacle state—Connected

2. Occupant state—Unconfigured

3. Condition—OK

Now the system is aware of the board, but not the usable devices which reside on the board.  Temperature is monitored and power and cooling affect the attachment point's condition. In order to transition to the Configured state, the operator must execute:

```
# cfgadm -c configure logical ap_id
```

The states and conditions should be:

1. Receptacle state—Connected

2. Occupant state—Configured

3. Condition—OK

Now the system is also aware of the usable devices which reside on the board and all devices may be mounted or configured to be used.

Should this operation fail for any reason, the states and conditions will still transition to configured. This creates a special situation where the situation is partially configured (or unconfigured). In this situation, only an unconfigure operation can be tried. An additional configure attempt to configure the partial portion is not permitted.


## *Enabling a Disabled I/O Board*

A board can be placed in the disabled board list. This basically excludes the board from being used as a result of cold start.

All other effects are the same as above after a successful insertion. Note that disabled boards persist even if a different board is subsequently placed in the same slot.

### Adding Storage

Additional storage may be added to a system by unconfiguring the board which is to be cabled to the additional storage and then reconfiguring the board. Before a board can be unconfigured, all active use of the devices must be terminated (see "Unconfiguring a Board" on page 11" below).

Adding storage can be accomplished by executing the following sequence:

```
# cfgadm -c unconfigure logical ap_id
(Physically add the storage devices.)
# cfgadm -c configure logical ap_id
```

Note that only the Occupant state should change. The Receptacle state and condition should remain the same.

## Examples of Removing Hardware

There are three degrees in which hardware is removed from a system:

- **Unconfigure a board**—remove a system's awareness of the usable hardware of a board. Temperature, power, and cooling are still monitored for the board but evidence of the usable devices is removed.
- **Disconnect the attachment point**—remove the environment awareness of the board and electrically power down the board.
- **Remove board**

### Unconfiguring a Board

Before Unconfiguring a Sunfire IO Board, you must manually terminate all usage of devices.

In order to unconfigure a board, the operator must ensure that all networking interfaces are down and all storage devices are unmounted and closed. DR does not automatically terminate network use or close all devices.

There currently is no way to ensure that the use of the network remains terminated or that all devices remain closed. Other clients may remount them between the time of the unmount and the unconfigure. The states and conditions must either be:

Receptacle state—Connected
Occupant state—Configured
Condition—OK

or

Receptacle state—Connected
Occupant state—Configured
Condition—Failing

Once unmounting is established, the operator can unconfigure a board by executing the following:

```
# cfgadm -c unconfigure logical ap_id
```

Environmental monitoring continues, but the usable devices are no longer known to the system. Should this operation fail for any reason, the states and conditions will remain the same as before. This creates a special situation where the situation is partially configured (or unconfigured depending on perspective). In this situation, only an unconfigure operation can be tried. A configure attempt to reconfigure the partial portion is not permitted.

## *Disconnecting the Attachment Point*

Disconnecting the attachment point can occur without restriction any time that the states and condition are either:

Receptacle state—Connected
Occupant state—Unconfigured
Condition—OK

or

Receptacle state—Connected
Occupant state—Unconfigured
Condition—Failing

If this operation fails, the situation is considered pathological and the system should either panic or shut down. If this operation succeeds, the states and conditions should be:

Receptacle state—Disconnected
Occupant state—Unconfigured
Condition—OK

or

Receptacle state—Disconnected
Occupant state—Unconfigured
Condition—Unknown

Note that the condition of a board disconnecting may persist until either a reconnect attempt or a board removal. A board which had the condition of failing prior disconnecting will transition to unknown.

To execute the disconnect transition, execute the following:

```
# cfgadm -c disconnect logical ap_id
```

After disconnecting a board, there is no longer any system monitoring of the board. Power is turned off and the board can basically be removed at any time.

## Removing a Board

Once in the disconnected state, the board can be removed. Note that the condition is reset upon removal:

Receptacle state—Empty
Occupant state—Unconfigured
Condition—Unknown

# Dynamic Reconfiguration Qualification Issues

This chapter describes how to determine what hardware components in an Sun Enterprise 3*x*00/4*x*00/5*x*00/6*x*00 server are suitable for using Dynamic Reconfiguration. In addition, this chapter discusses DR configuration issues.

## Site Qualification for DR

Before you can use DR with an Sun Enterprise 3*x*00-6*x*00 server, you must evaluate how currently installed hardware components are being used. The two criteria for deciding whether DR is usable include:

- Devices must not be in use by the operating system
- Drivers must be detachable or suspendable

### Devices Not in Use

A device cannot be unconfigured/disconnected while it is still being used by the Solaris software. This requirement means that disks attached to an I/O board should be unmounted before any attempt is made to unconfigure/disconnect that board. Any attempt to unconfigure/disconnect a board whose devices are still in use will generate an error.

# Device Driver Suspendability

Suspendability means device drivers have the ability to execute the `sysctrl_quiesce()` function. Externally, this provides the ability to:

- Stop user threads
- Execute the `DDI_SUSPEND` call in each device driver
- Stop the clock
- Stop the CPUs

Inability to accomplish these actions (primarily the first two) cause a system to not be suspendable. The indirect effect of these capabilities is to stop transactions on the interconnect.

Drivers that are suspendable are suspend-safe drivers. Suspend-unsafe drives are not suspendable.

# Device Driver Detachability

The attributes that make devices detachable are:

- Drivers that support `DDI_DETACH`
- Devices arranged so that they can be detached

Assuming that a site is using a set of detach-qualified drivers, the major factor for determining detachability is the system configuration—the arrangement of boards, slots, and connectors.

Drivers that are detachable are detach-safe drivers. Detach-unsafe drivers are not detachable.

## Arrangement of Devices for Detachability

To be detachable, boards must be arranged so that critical resources can be accessed through alternate pathways.

Imagine an I/O board that has two disk chains, one for the boot path and one for a set of secondary data. If both disk chains are connected to the same I/O board, then that board is not detachable because you cannot disconnect a board when critical resources are connected to it. Similarly, the CPU board cannot be detached from a single-CPU system.

Two solutions exist to solve this problem:

- Put the second disk chain on a separate I/O board
- Add a second *path* to the device through a second I/O board

In the first case, the secondary I/O board can be detached (with a loss of access to the secondary disk chain). In the second case, the I/O board can be detached (using Alternate Pathing software to switch access through the alternate board) *without* losing access to the secondary disk chain.

# Setup of Dynamic Reconfiguration

Before administrators perform a configure/unconfigure operation, they must set up the component intended for the configure/unconfigure operation.

## Setup for DR Configure

Very little setup is required for adding resources.

### I/O Devices

Setup for a DR Configure requires attention to one isse: To prevent disk controller re-numbering when choosing the slot into which a board is inserted. See "Disk Devices" on page 19 for information about avoiding disk controller renumbering.

## Setup for DR Unconfigure

This section describes how to set up DR before performing an Unconfigure operation.

### I/O and Network Devices

In order to ensure a board that hosts one or more I/O controllers attached to vital system resources can change to the unconfigured state, use the Alternate Pathing software. If, for example, the root or /usr disk partition is on a disk connected to a controller on the board, the board cannot be unconfigured unless there is a hardware alternate path to the disk or the disk is mirrored. The alternate path and/or the mirrors must be hosted by other I/O controllers in the machine. The same applies to network controllers. The board that hosts the Ethernet controller cannot be detached unless an alternate path exists to an Ethernet controller on another board for this network connection.

The system swap space should be configured as multiple partitions on disks attached to controllers hosted by different boards. With this kind of configuration, a particular swap partition is not a vital resource because swap partitions can be added and deleted dynamically. See swap(1M) for more information.

---

**Note –** When memory (swapfs) or disk swap space is detached, there must be enough memory or swap disk space remaining in the machine to accommodate currently running programs.

---

A board hosting non-vital system resources can be unconfigured whether or not there are alternate paths to the resources. All of its file systems must be unmounted and its swap partitions must be deleted. You may have to kill processes that have open files or devices, or place a hard lock on the file systems (using lockfs(1M)) before unmounting them. All I/O device drivers must be detachable.

# Reconfiguration After a DR Operation

This section describes how to reconfigure your system after you have configured or unconfigured a system board.

## When to Reconfigure

You might need to reconfigure the system under several conditions, including:

- **Board Addition**—When adding a board, you must execute the reconfiguration sequence to configure the I/O devices associated with the board.
- **Board Deletion**—If you remove a board that is not to be replaced, you may (but do not have to) execute the reconfiguration sequence to clean up the /dev links for disk devices.
- **Board Replacement**—If you remove a board then reinsert it into a different slot, or replace a board with another board that has different I/O devices, you must execute the reconfiguration sequence to configure the I/O devices associated with the board. However, if you replace a board with another board that hosts the *same* set of I/O devices, inserting the replacement into the *same* slot, you do not need to execute the reconfiguration sequence. But be sure to insert a replacement into the same slot that was vacated to retain the original /dev link names.

# I/O Devices

The DR reconfiguration sequence is the same as the Solaris reconfiguration boot sequence (`boot -r`):

```
drvconfig; devlinks; disks; ports; tapes;
```

When the reconfiguration sequence is executed after a board is configured, device path names not previously seen by the system are entered into the /etc/path_to_inst file. The same path names are also added to the /devices hierarchy and links to them are created in the /dev directory.

# Disk Devices

The disks(1m) utility creates symbolic links in the /dev/dsk and /dev/rdsk directories pointing to the actual special disk device files under the /devices directory tree. These entries take the form /dev/dsk/cXtXdXsX where:

- c*X* is the disk controller number
- t*X* corresponds to the disk target number, in most cases
- d*X* refers to the logical unit number
- s*X* is the partition number

When the reconfiguration sequence is executed after a board is unconfigured/disconnected, the /dev links for all the disk partitions on that board are deleted. The remaining boards retain their current numbering. Disk controllers on a newly inserted board are assigned the next available lowest number by disks(1M).

Removing boards that contain one or more disk controllers prompts the disks(1m) utility to examine entries in /dev/dsk and /dev/rdsk. These entries list the disks attached to the removed controller(s). The disks(1m) utility discovers that references to the disconnected devices have been removed from /dev/dsk and /dev/rdsk. This removal action makes the logical controller numbers available for re-use. This re-use of controller numbers can lead to non-intuitive controller numbers being assigned to disk controllers that are added to the system.

---

**Note –** The disk controller number is part of the /dev link name used to access the disk. If that number changes during the reconfiguration sequence, the /dev link name also changes. This change may affect file system tables and software, such as Solstice DiskSuite, which uses the /dev link names. Update /etc/vfstab files and execute other administrative actions necessary due to the changes in the /dev link names.

---

# DR and AP Interaction

DR and Alternate Pathing (AP) do not directly interact. Alternate Pathing is treated as device-level management on Sun Enterprise 3000-6000 servers. For more information, see the *Enterprise Server Alternate Pathing 2.1 User's Guide*. Using the Alternate Pathing software increases the probability that a system board can be disconnected using the DR software.

# RPC Timeout or Loss of Connection

RPC time-outs occur by default after two minutes. Administrators may need to increase this time-out value to avoid time-outs during a DR-induced operating system quiescence, which may take longer than two minutes. These changes affect both the client and server machines.

# Operating System Quiescence

During a DR unconfigure/disconnect operation on a system board with non-pageable Open Boot PROM (OBP) or kernel memory, the operating system is briefly paused, which is known as operating system quiescence; that is, all operating system and device activity on the backplane must cease for a few seconds during a critical phase of the operation.

Before it can achieve quiescence, the operating system must temporarily suspend all processes, processors, and device activities. If the operating system cannot achieve quiescence, it displays its reasons, which may include the following:

- A user thread did not suspend
- Real-time processes are running
- A device exists that cannot be paused by the operating system

The conditions that cause processes to fail to suspend are generally temporary in nature. You can retry the operation until the quiescence succeeds.

If the operating system fails to achieve quiescence, pay close attention to the reasons for the failure. If the operating system encountered a transient condition—a failure to suspend a process—you can try the operation again.

# Suspend-Safe/Suspend-Unsafe Devices

A suspend-safe device is one that does not access memory or interrupt the system while the operating system is in quiescence. A driver is considered suspend-safe if it supports operating system quiescence (suspend/resume). It also guarantees that when a suspend request is successfully completed, the device that the driver manages will not attempt to access memory, even if the device is open when the suspend request is made. All other I/O devices are suspend-unsafe when open.

The `cfgadm -x "quiesce-test"` command returns a list of all suspendable drivers.

The operating system refuses a quiescence request if a suspend-unsafe device is open. To manually suspend the device, you may have to close the device by killing the processes that have it open, ask users not to use the device, or disconnect the cables. For example, if a device that allows asynchronous unsolicited input is open, you can disconnect its cables prior to activating operating system quiescence and reconnect them after the operating system resumes. Doing this prevents traffic from arriving at the device and, thus, the device has no reason to access the backplane.

# Special Quiescence Handling for Tape Devices

The sequential nature of tape devices prevents them from being reliably suspended in the middle of an operation, and then resumed. Therefore, all tape drivers are suspend-unsafe and cannot be made quiescent. Before executing a DR operation that activates operating system quiescence, make sure all tape devices are closed or not in use.

# DR and Detach-Unsafe Devices

Not all drivers support the Sun Enterprise 3*x*00-6*x*00 system's Dynamic Reconfiguration feature.

# Removing Detach-Unsafe Drivers

Administrators can remove some otherwise undetachable drivers manually.

1. **Stop all usage of the controller for the detach-unsafe device, and all other controllers of the same type on all boards in the machine.**

   Since the detach-unsafe driver must be unloaded using the `modunload` command (next step), you must stop use of that controller type on *all* system boards. The remaining controllers can be used again after the DR Unconfigure is complete.

2. **Use standard Solaris interfaces to manually close all such drivers on the board and use the `modunload` command to unload them.**

   See `modload(1M)` in *Platform Notes: Sun Enterprise 3000, 4000, 5000, and 6000 Systems.*

3. **Detach the system board in the normal fashion.**

   If you cannot execute the above steps, you may reboot after adding the board to the disabled board list using the NVRAM setting `disabled-board-list` (see Platform Notes) and the board can later be removed.

---

**Note –** Many third-party drivers (those purchased from vendors other than Sun Microsystems) do not properly support the standard Solaris software `modunload` interface. Conditions that invoke the functions occur infrequently during normal operation and the functions are sometimes missing or work improperly. Test these driver functions during the qualification and installation phases of any third-party device.

---

## Adding Suspend-Safe Devices

With the 2.6 release, Solaris provides a new interface for building suspend-safe devices. This interface is described in the *Solaris 2.6 Reference Manual Answerbook,* under DDI and DKI Driver Entry Points. This interface also describes how to create suspend-bypass devices.

# Using Dynamic Reconfiguration

This chapter describes how to configure and unconfigure system boards.

- Configuring a System Board—page 23
- Preparing to Unconfigure a Board—page 25
- Unconfiguring a System Board—page 25
- Viewing System Information—page 28
- Troubleshooting—page 30

# Configuring a System Board

**Note –** This section gives a broad overview of the actions that occur when you execute DR configure. For step-by-step instructions, see "To Configure a Board With cfgadm" on page 24.

Administrators can configure system boards that are present in the machine, powered on, and not being used by the operating system. These unconfigured boards may have been hot swapped into the system after the system was booted, disabled by the NVRAM setting `disable-board-list` when the system was booted, or previously unconfigured. See "Site Qualification for DR" on page 15 for information on what board setups can be configured.

**Note –** The operator is responsible for ensuring that a compatible and working board is plugged in to the system.

# To Configure a Board

The two-step process of configuring a board involves connecting the attachment point and configuring its occupant. The `cfgadm(1M)` command can perform both steps at once.

## ▼ To Configure a Board With `cfgadm`

The `cfgadm` command was introduced in Chapter 1. A quick reference guide is available in the `cfgadm(1M) man` page.

1. **Connect all peripheral cables after inserting the board and before executing the following commands.**

2. **Change the state of the attachment point to Connected with a command of the form:**

```
# cfgadm -v -c connect sysctrlx:slotx
```

You can skip this step in most cases.

3. **Change the state of the occupant to Configured with a command of the form:**

```
# cfgadm -v -c configure sysctrlx:slotx
```

This command will also connect the receptacle if you omit step 2.

4. **Reconfigure the devices on the board by entering:**

```
# drvconfig; devlinks; disks; ports; tapes;
```

⚠ **Caution –** If your system uses Alternate Pathing software, ensure that the devices on the board are active before using AP commands (such as `apconfig`). You can verify the devices are active and available by running `format`, `prtvtoc`, and similar commands on the target drives and controllers.

# Unconfiguring a System Board

This section gives a broad overview of the actions that occur when you execute DR unconfigure.

System boards that are currently being used by the operating system can be unconfigured if they meet the requirements covered in "Setup for DR Unconfigure" on page 17.

---

**Note –** The screen, mouse, and keyboard will not be operational while the system is suspended.

---

## Preparing to Unconfigure a Board

Before the Unconfigure operation can be completed, you must terminate all usage of board resources (I/O devices). You must manually terminate the use of all I/O devices, including network interfaces.

---

**Note –** To identify the components that are on the board to be unconfigured, use the `ifconfig`, `mount`, `pf`, or `swap` commands. Another somewhat less informative way is to execute the `prtdiag(1M)` command.

---

### Network Devices

DR does not automatically terminate use of all network interfaces on the board that is being disconnected. You must manually terminate the use of each interface.

DR does not allow an Unconfigure operation on any interface that fits any of the following conditions. In these cases, the Unconfigure operation fails and DR displays an error message. The operation fails if the interface is:

■ The primary network interface for the machine; that is, the interface whose IP address corresponds to the network interface name contained in the file `/etc/nodename`. Note that bringing down the primary network interface for the machine prevents network information name services from operating, which results in the inability to make network connections to remote hosts using applications such as `ftp(1)`, `rsh(1)`, `rcp(1)`, `rlogin(1)`. NFS client and server operations are also affected.

- The active alternate for an Alternate Pathing (AP) meta device when the AP meta device is plumbed. Interfaces used by the AP system should not be the active path when the board is being unconfigured. Manually switch the active path to one that is not on the board being unconfigured. If no such path exists, manually execute the `ifconfig down` and `ifconfig unplumb` commands on the AP interface. (To manually switch an active path, use the `apconfig(1M)` command.)

## I/O Devices

All I/O devices must be closed before they are unconfigured. To see which processes have these devices open, use the `fuser(1M)` command.

You must perform the following tasks for I/O devices.

- If the redundancy features of Alternate Pathing or Solstice DiskSuite mirroring are used to access a device connected to the board, reconfigure these subsystems so that the device or network is accessible by way of controllers on other system boards.
- Unmount file systems, including Solstice DiskSuite meta-devices that have a board resident partition. (Example: `umount /partit`)
- Remove Solstice DiskSuite or Alternate Pathing databases from board-resident partitions. The location of Solstice DiskSuite or Alternate Pathing databases is explicitly chosen by the user and can be changed.
- Remove any private regions used by Sun Volume Manager or Veritas Volume Manager. Volume Manager by default uses a private region on each device that it controls, so such devices must be removed from Volume Manager control before they can be detached.
- Any RSM 2000 controllers on the board that is being detached should be taken offline, using the `rm6` or `rdacutil` commands.
- Remove disk partitions from the swap configuration.
- Either kill any process that directly opens a device or raw partition, or direct it to close the open device on the board.
- If a detach-unsafe device is present on the board, close all instances of the device and use `modunload(1M)` to unload the driver.

**Caution –** Unmounting file systems may affect NFS client systems.

## Finishing the Unconfigure Operation

Once all board usage is terminated, you can perform the complete Unconfigure operation. If a device is still in use at this time, the Unconfigure operation fails and the device in use is reported. After resolving the problem, you can perform the complete Unconfigure operation again.

The complete Unconfigure operation may also fail due to quiescence problems, which are described in "Operating System Quiescence" on page 20. After resolving the quiescence problem you can again execute the complete Unconfigure operation.

If you decide that you do not want to proceed with the Unconfigure operation at this time, you can abort the Unconfigure. The board's memory is returned to normal usage and detached board devices are reattached. If the system configuration was modified to remove board usage (that is, file systems were unmounted and networks were unplumbed), it is the administrator's responsibility to undo these modifications and return the devices to normal operation.

You can now remove the board using hot swap, leave it in the system unconfigured, or configure it at a later time.

# ▼ To Unconfigure a Board with `cfgadm`

Before executing the following steps, see "Preparing to Unconfigure a Board" on page 25. The basic concepts are not repeated in this section.

1. **Make sure the device is not being used.**

2. **Put the attachment point occupant into the Unconfigured state with a command of the general form:**

```
# cfgadm -v -c unconfigure sysctrlx:slotx
```

3. **Put the attachment point into the Disconnected state with a command of the general form:**

```
# cfgadm -v -c disconnect sysctrlx:slotx
```

4. **Remove the board.**

# Viewing System Information

The cfgadm program allows you to display information about the suspend-unsafe devices as well as information about the board selected during DR operations. For cfgadm, this information is accessible when the command is issued with no arguments. An expanded display is available using the −v (verbose) switch.

In addition the −l switch lists the state and condition of each DR attachment point.

## ▼ To View Current Configurable Hardware Information

● **Execute the command,** cfgadm**.**

When used without the -l switch, the Solaris software displays information about all known DR attachment points:

**TABLE 3-1**    cfgadm Information Display

| Compid | State | Condition |
|---|---|---|
| sysctrl0:slot0 | configured | ok |
| sysctrl0:slot2 | configured | ok |
| sysctrl0:slot3 | configured | ok |
| sysctrl0:slot4 | unconfigured | unknown |
| sysctrl0:slot5 | configured | failing |
| sysctrl0:slot6 | configured | ok |
| sysctrl0:slot7 | configured | ok |
| sysctrl0:slot8 | disabled | unusable |

# ▼ To View Expanded Hardware Information

● **Execute the command,** `cfgadm -v -l`.

The Solaris software displays the following:

**TABLE 3-2**  Expanded `cfgadm` Information Display

| Compid | State When | Condition | Type | Information |
| --- | --- | --- | --- | --- |
| sysctrl0:slot0 | configured Nov 5 | ok | CPU | 2 SUNW, UltraSPARC, 168 MHz |
| sysctrl0:slot2 | configured Nov 5 | ok | MEMORY | 512mb, 2 way interleaved |
| sysctrl0:slot3 | configured Nov 5 | ok | IO | PCI |
| sysctrl0:slot4 | configured Nov 5 | test | MEMORY | 512mb 2 way interleaved |
| sysctrl0:slot5 | unconfigured Nov 5 | ok | IO | PCI |

# Troubleshooting

## Diagnostic Messages

The `cfgadm` command returns diagnostic messages in the standard error output command tool window. Other than syntax errors, these messages include:

`cfgadm: Configuration administration not supported on this machine`

`cfgadm:` *operation*`: configuration operation not supported on this machine`

`cfgadm:` *operation*`: Insufficient privileges`

`cfgadm: hardware component is busy, try again`

`cfgadm: System is busy, try again`

`cfgadm:` *operation*`: Operation requires a service interruption`

`cfgadm:` *operation*`: Data error:` *error_text*

`cfgadm:` *operation*`: Hardware specific failure:` *error_text*

See `config_admin(3X)` for additional error message detail.

## Troubleshooting Specific Failures

There are several common types of failure:

- Inability to unconfigure
- Inability to configure

### Unable to Perform an Unconfigure Operation

An unconfiguration may fail if an I/O board has a busy or open device. Note that in such a situation, the board is left only partially unconfigured. The operation stopped at the busy device.

To regain access to the devices which were not unconfigured, the board must be completely unconfigured and then reconfigured.

In such a case, the system will log messages similar to the following:

```
NOTICE: unconfiguring dual-pci board in slot 7
NOTICE: dual-pci board in slot 7 partially unconfigured reason:
sysc iohelp unconfigure: Device busy
output from sysctrl unconfigure is:
detach failed: /pci@f,4000/SUNW,isptwo@3/sd@2,0 is busy
```

To continue the unconfigure operation, unmount the device and retry the unconfigure operation. Attempting to configure this board again without first successfully transitioning to the unconfigured state is not allowed.

## Unable to Perform a Configure Operation

A configure operation may fail because an I/O board with a device does not currently support hotplugging. In such a situation, the board is now only partially configured. The operation has stopped at the unsupported hotplug device. In this situation, the board must be brought back to the unconfigured state before another configure attempt. In such a case the system will log messages similar to the following:

```
NOTICE: configuring dual-sbus-soc+ board in slot 4
NOTICE: dual-sbus-soc+ board in slot 4 partially configured
        reason:
sysc iohelp configure: Bad address
output form sysctrl configure is:
attach failed: /sbus@8,0/SUNW,foo@d,10000/bar
```

To continue the configure operation, either remove the unsupported device's driver or replace it with a new version of the driver that will support hotplugging.

# Glossary

**AP** See Alternate Pathing.

**Alternate Pathing** Alternate Pathing (AP) is software package which allows the use of multiple paths (connections) between a server and a SPARCstorage array (SSA) or a network. If one path fails, AP can ensure that the SSA or network is still available through the alternate path. For example, the alternate path can be a second port on an interface board, or an entirely separate interface board. See also Dynamic Reconfiguration.

**Attachment point** Any point within a system where hardware resources can be added to or subtracted from the machine. DR support and hot-plug support are required for that point. An attachment point has two parts, receptacle and occupant.

`cfgadm` **command** See the `cfgadm` man page for more information about the configuration administration command.

**Commands for DR** The common commands include: `cfgadm`. Refer to the man pages for information about the use of this command. Refer to the release notes for any late-breaking news of other commands.

**Configuration** The collection of attached devices known to the system. The system cannot use a physical device may xxxx until the configuration is updated.

**Connection** A board is inserted and electrically connected and powered up.

**Detachability** The device driver supports `DDI_DETACH` and the device (such as an I/O board or a SCSI chain) is physically arranged so that it can be detached.

**Disconnection** A board is inserted but electrically disconnectedand can be unplugged at any time.

**DR** See Dynamic Reconfiguration

**Dynamic Reconfiguration** Dynamic Reconfiguration (DR) is software which allows the administrator to (1) view a system configuration, (2) suspend or restart operations involving a port, storage device, or board, and (3) reconfigure the system (detach or attach

hot-swappable devices such as disk drives or interface boards) without the need to power down the system. When DR is used with Alternate Pathing or Solstice DiskSuite software (and redundant hardware), the server can continue to communicate with disk drives and networks without interruption while a service provider replaces an existing device or installs a new device. DR supports replacement of a CPU/Memory, provided the memory on the board is not interleaved with memory on other boards in the system.

| | |
|---|---|
| **Hot Swap** | A hot swap device has special DC power connectors and logic circuitry that allow the device to be inserted without the necessity of turning off the system. |
| **Receptacle** | A receiver such as a board slot or SCSI chain. |
| **Suspendability** | To be suitable for DR, a device driver must have the ability to stop user threads, execute the DDI_SUSPEND call, stop the clock, and stop the CPUs. |
| **Occupant** | Hardware resource such as a system board or a disk drive that occupies a DR receptacle or slot. |

# Index