



---

## Platform Notes: Sun Enterprise 250 Server

---

Sun Microsystems, Inc.  
901 San Antonio Road  
Palo Alto, CA 94303-4900  
U.S.A

Part No.: 805-4453-10  
October 1998 , Revision A

Send comments about this document to:  
[docfeedback@sun.com](mailto:docfeedback@sun.com)

1998 Sun Microsystems, Inc., 901 San Antonio Road, Palo Alto, California 94303-4900 U.S.A.

This product or document is protected by copyright and distributed under licenses restricting its use, copying, distribution, and decompilation. No part of this product or document may be reproduced in any form by any means without prior written authorization of Sun and its licensors, if any. Third-party software, including font technology, is copyrighted and licensed from Sun suppliers.

Parts of the product may be derived from Berkeley BSD systems, licensed from the University of California. UNIX is a registered trademark in the U.S. and other countries, exclusively licensed through X/Open Company, Ltd.

Sun, Sun Microsystems, the Sun logo, Java, the Java Coffee Cup logo, and Solaris are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and in other countries. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. in the U.S. and other countries. Products bearing SPARC trademarks are based upon an architecture developed by Sun Microsystems, Inc.

The OPEN LOOK and Sun™ Graphical User Interface was developed by Sun Microsystems, Inc. for its users and licensees. Sun acknowledges the pioneering efforts of Xerox in researching and developing the concept of visual or graphical user interfaces for the computer industry. Sun holds a non-exclusive license from Xerox to the Xerox Graphical User Interface, which license also covers Sun's licensees who implement OPEN LOOK GUIs and otherwise comply with Sun's written license agreements.

RESTRICTED RIGHTS: Use, duplication, or disclosure by the U.S. Government is subject to restrictions of FAR 52.227-14(g)(2)(6/87) and FAR 52.227-19(6/87), or DFAR 252.227-7015(b)(6/95) and DFAR 227.7202-3(a).

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.

Copyright 1998 Sun Microsystems, Inc., 901 San Antonio Road, Palo Alto, California 94303 Etats-Unis. Tous droits réservés.

Ce produit ou document est protégé par un copyright et distribué avec des licences qui en restreignent l'utilisation, la copie, la distribution, et la décompilation. Aucune partie de ce produit ou document ne peut être reproduite sous aucune forme, par quelque moyen que ce soit, sans l'autorisation préalable et écrite de Sun et de ses bailleurs de licence, s'il y en a. Le logiciel détenu par des tiers, et qui comprend la technologie relative aux polices de caractères, est protégé par un copyright et licencié par des fournisseurs de Sun.

Des parties de ce produit pourront être dérivées des systèmes Berkeley BSD licenciés par l'Université de Californie. UNIX est une marque déposée aux Etats-Unis et dans d'autres pays et licenciée exclusivement par X/Open Company, Ltd.

Sun, Sun Microsystems, le logo Sun, Java, le logo Java Coffee Cup, et Solaris sont des marques de fabrique ou des marques déposées, ou marques de service, de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays. Toutes les marques SPARC sont utilisées sous licence et sont des marques de fabrique ou des marques déposées de SPARC International, Inc. aux Etats-Unis et dans d'autres pays. Les produits portant les marques SPARC sont basés sur une architecture développée par Sun Microsystems, Inc.

L'interface d'utilisation graphique OPEN LOOK et Sun™ a été développée par Sun Microsystems, Inc. pour ses utilisateurs et licenciés. Sun reconnaît les efforts de pionniers de Xerox pour la recherche et le développement du concept des interfaces d'utilisation visuelle ou graphique pour l'industrie de l'informatique. Sun détient une licence non exclusive de Xerox sur l'interface d'utilisation graphique Xerox, cette licence couvrant également les licenciés de Sun qui mettent en place l'interface d'utilisation graphique OPEN LOOK et qui en outre se conforment aux licences écrites de Sun.

CETTE PUBLICATION EST FOURNIE "EN L'ETAT" ET AUCUNE GARANTIE, EXPRESSE OU IMPLICITE, N'EST ACCORDEE, Y COMPRIS DES GARANTIES CONCERNANT LA VALEUR MARCHANDE, L'APTITUDE DE LA PUBLICATION A REPENDRE A UNE UTILISATION PARTICULIERE, OU LE FAIT QU'ELLE NE SOIT PAS CONTREFAISANTE DE PRODUIT DE TIERS. CE DENI DE GARANTIE NE S'APPLIQUERAIT PAS, DANS LA MESURE OU IL SERAIT TENU JURIDIQUEMENT NUL ET NON AVENU.



Please  
Recycle



Adobe PostScript

# Preface

---

*Platform Notes: Sun Enterprise 250 Server* contains the following information for system administrators and advanced users of Sun Enterprise 250 servers:

- OpenBoot™ commands and variables for configuring various aspects of system behavior
- Detailed software-related procedures for adding, removing, or replacing hot-pluggable disk drives
- Procedures for mapping between the logical and physical device names of internal storage devices

---

## 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 Handbook for Sun Peripherals*
- AnswerBook™ online documentation for the Solaris™ 2.x software environment
- Other software documentation that you received with your system

---

# Shell Prompts

The following table shows the default system prompt and superuser prompt for the C shell, Bourne shell, and Korn shell.

**TABLE P-1** Shell Prompts

<b>Shell</b>	<b>Prompt</b>
C shell	<i>machine_name%</i>
C shell superuser	<i>machine_name#</i>
Bourne shell and Korn shell	\$
Bourne shell and Korn shell superuser	#

---

# Typographic Conventions

TABLE P-2 Typographic Conventions

Typeface or Symbol	Meaning	Examples
AaBbCc123	The names of commands, files, and directories; on-screen computer output.	Edit your <code>.login</code> file. Use <code>ls -a</code> to list all files. % You have mail.
<b>AaBbCc123</b>	What you type, when contrasted with on-screen computer output.	% <b>su</b> Password:
<i>AaBbCc123</i>	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 <i>User's Guide</i> . These are called <i>class</i> options. You <i>must</i> be <i>root</i> to do this. To delete a file, type <code>rm filename</code> .

---

## Related Documentation

TABLE P-3 Related Documentation Titles

Application	Title
Hardware installation and service	<i>Sun Enterprise 250 Server Owner's Guide</i>
System and network administration	<i>Solaris System Administrator AnswerBook</i>
Miscellaneous	<i>Solaris on Sun Hardware AnswerBook</i> <i>Solaris Handbook for Sun Peripherals</i> <i>Solaris 7 Sun Hardware Platform Guide</i>

---

## Sun Documentation on the Web

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

`http://docs.sun.com`.

---

## Sun Welcomes Your Comments

We are interested in improving our documentation and welcome your comments and suggestions. You can email your comments to us at:

`docfeedback@sun.com`.

Please include the part number of your document in the subject line of your email.

# Contents

---

<b>Preface</b>	<b>iii</b>
<b>1. System Configuration Parameters</b>	<b>1</b>
UPA Probing	2
PCI Probing	2
Environmental Monitoring and Control	4
Automatic System Recovery	5
“Soft” Deconfiguration via Status Property	5
“Hard” Deconfiguration	5
ASR User Override Capability	6
Auto-Boot Options	7
Reset Scenarios	8
<b>2. Disk Drive Hot-Plug Procedures</b>	<b>11</b>
Overview	11
Adding a Hot-Pluggable Disk Drive	12
Configuring the New Disk Drive Within Your Application	15
Configuring the New Disk Drive for a UNIX File System (UFS)	16
Adding a Disk to a Solstice DiskSuite Disk Set	17
Replacing a Faulty Hot-Pluggable Disk Drive	17

Prepare Spare Drives	17
Identifying the Faulty Disk Drive	18
Replacing the Disk Drive Within Your Application	19
UNIX File System (UFS)	19
Solstice DiskSuite	22
Removing a Hot-Pluggable Disk Drive	26
Identifying the Faulty Disk Drive	27
Removing a Disk Drive From Your Application	28
UNIX File System (UFS)	28
Solstice DiskSuite	30
<b>3. Mapping Between Logical and Physical Device Names</b>	<b>33</b>
Overview	33
Mapping From Error Message to Disk Slot Number and UNIX Logical Name	34
Mapping From UNIX Logical Name to Disk Slot Number	36
Mapping From Disk Slot Number to UNIX Logical Name	37



# System Configuration Parameters

---

This chapter describes the NVRAM configuration variables and OpenBoot PROM (OBP) commands available for configuring the following aspects of Sun Enterprise 250 system behavior:

- “UPA Probing
- “PCI Probing
- “Environmental Monitoring and Control
- “Automatic System Recovery

NVRAM configuration variables covered in this chapter include:

- `upa-port-skip-list`
- `pci0-probe-list`
- `pci-slot-skip-list`
- `env-monitor`
- `asr-disable-list`
- `auto-boot-on-error?`
- `diag-trigger`

OBP commands covered in this chapter include:

- `asr-enable`
- `asr-disable`
- `.asr`

---

# UPA Probing

Sun Enterprise 250 servers, like all UltraSPARC™-based systems, are based on the high-speed Ultra Port Architecture (UPA) bus, a switched system bus that provides up to 32 port ID addresses (or slots) for high-speed motherboard devices like CPUs, I/O bridges, and frame buffers. The Enterprise 250 server provides up to three active ports for the following subsystems.

TABLE 1-1 UPA System Bus Ports

Device Type	UPA Slot	Physical Implementation
CPU	0-1	Two plug-in slots
UPA-PCI bridge	1f	Soldered on motherboard

The order of probing these three port IDs is *not* subject to user control; however, a list of ports *can be excluded* from probing via the `upa-port-skip-list` NVRAM variable. In the following example, the `upa-port-skip-list` variable is used to exclude CPU-1 from the UPA probe list.

```
ok setenv upa-port-skip-list 1
```

This capability lets you exclude a given device from probing (and subsequent use) by the system without physically removing the plug-in card. This can be useful in helping to isolate a failing card in a system experiencing transient failures.

---

# PCI Probing

Of the Enterprise 250 server's two PCI buses, Bus 0 (`/pci@1f,4000` in the device tree) is unique in that it is the only PCI bus that contains motherboard (non plug-in) devices such as the on-board SCSI controller. By definition, such devices cannot be unplugged and swapped to change the order in which they are probed. To control the probing order of these devices, the system provides the NVRAM variable

`pci0-probe-list`. This variable controls both the probing order and exclusion of devices on PCI Bus 0. The values that you can specify in the `pci0-probe-list` are defined in the following table.

**TABLE 1-2** Probe Values

PCI Device Number	Function
2	PCI slot 2 (as labeled on system rear panel)
3	On-board SCSI controller for internal and external SCSI devices
4	PCI slot 1 (as labeled on system rear panel)
5	PCI slot 0 (as labeled on system rear panel)

**Note** – The values in this list are based on the PCI device number and do *not* refer to the back panel slot numbering scheme of 0–3.

In the following example, the `pci0-probe-list` variable is used to define a probing order of 5–2–4, while excluding from the probe list the on-board SCSI controller for internal and external SCSI devices.

```
ok setenv pci0-probe-list 5,2,4
```

Note that the `pci0-probe-list` variable has no effect on probing of the top PCI slot (slot 3 on the system rear panel). However, another NVRAM variable, `pci-slot-skip-list`, is available for excluding any PCI slot from the PCI probe list. In the following example, the `pci-slot-skip-list` variable is used to exclude back panel slots 0 and 3 from the PCI probe list.

```
ok setenv pci-slot-skip-list 0,3
```

**Note** – The values in the `pci-slot-skip-list` correspond to the back panel numbering scheme of 0–3. If a PCI slot number appears in this list, it will be excluded from probing even if it appears in the `pci0-probe-list` variable.

---

# Environmental Monitoring and Control

Environmental monitoring and control capabilities for Enterprise 250 servers reside at both the operating system level and the OBP firmware level. This ensures that monitoring capabilities are operational even if the system has halted or is unable to boot. The way in which OBP monitors and reacts to environmental over temperature conditions is controlled by the NVRAM variable `env-monitor`. The following table shows the various settings for this variable and the effect each setting has on OBP behavior. For additional information about the system's environmental monitoring capabilities, see

About Reliability, Availability, and Serviceability Features

in the *Sun Enterprise 250 Server Owner's Guide*.

**TABLE 1-3** Environmental Monitor Values and Effects

Setting	Monitor Active?	Action Taken
<code>enabled</code> ( <i>default</i> )	Yes	In response to a critical over temperature condition, OBP issues a warning and automatically shuts down the system after 30 seconds.
<code>advise</code>	Yes	OBP issues a warning only, without shutting down the system.
<code>disabled</code>	No	OBP takes no action at all; environmental monitoring at the OBP level is disabled.

In the following example, the `env-monitor` variable is used to disable environmental monitoring at the OBP level.

```
ok setenv env-monitor disabled
```

---

**Note** – This NVRAM variable does not affect the system's environmental monitoring and control capabilities while the operating system is running.

---

---

# Automatic System Recovery

The automatic system recovery (ASR) feature allows Enterprise 250 servers to resume operation after experiencing certain hardware faults or failures. Power-on self-test (POST) and OpenBoot Diagnostics (OBDiag) can automatically detect failed hardware components, while an auto-configuring capability designed into the OBP firmware allows the system to deconfigure failed components and restore system operation. As long as the system is capable of operating without the failed component, the ASR features will enable the system to reboot automatically, without operator intervention. Such a “degraded boot” allows the system to continue operating while a service call is generated to replace the faulty part.

If a faulty component is detected during the power-on sequence, the component is deconfigured and, if the system remains capable of functioning without it, the boot sequence continues. In a running system, certain types of failures (such as a processor failure) can cause an automatic system reset. If this happens, the ASR functionality allows the system to reboot immediately, provided that the system can function without the failed component. This prevents a faulty hardware component from keeping the entire system down or causing the system to crash again.

## “Soft” Deconfiguration via Status Property

To support a degraded boot capability, the OBP uses the IEEE 1275 Client Interface (via the device tree) to “mark” devices as either *failed* or *disabled*, by creating an appropriate “status” property in the corresponding device tree node. By convention, UNIX will not activate a driver for any subsystem so marked.

Thus, as long as the failed component is electrically dormant (that is, it will not cause random bus errors or signal noise, etc.), the system can be rebooted automatically and resume operation while a service call is made.

## “Hard” Deconfiguration

In two special cases of deconfiguring a subsystem (CPUs and memory), the OBP actually takes action beyond just creating an appropriate “status” property in the device tree. At the first moments after reset, the OBP must initialize and functionally configure (or bypass) these functions in order for the rest of the system to work correctly. These actions are taken based on the status of two NVRAM configuration variables, `post-status` and `asr-status`, which hold the override information supplied either from POST or via a manual user override (see “ASR User Override Capability” on page 6).

## CPU Deconfiguration

If any CPU is marked as having failed POST, or if a user chooses to disable a CPU, then the OBP will set the Master Disable bit of the affected CPU, which essentially turns it off as an active UPA device until the next power-on system reset.

## Memory Deconfiguration

Detecting and isolating system memory problems is one of the more difficult diagnostic tasks. This problem is further complicated by the possibility of installing different capacity DIMMs within the same memory bank. (Each memory bank must contain four DIMMs of the same capacity.) Given a failed memory component, the firmware will deconfigure the entire bank associated with the failure.

## ASR User Override Capability

While the default settings will properly configure or deconfigure the server in most cases, it is useful to provide advanced users with a manual override capability. Because of the nature of “soft” versus “hard” deconfiguration, it is necessary to provide two related but different override mechanisms.

### “Soft” Deconfigure Override

For any subsystem represented by a distinct device tree node, users may disable that function via the NVRAM variable `asr-disable-list`, which is simply a list of device tree paths separated by spaces.

```
ok setenv asr-disable-list /pci@1f,2000 /pci@1f,4000/scsi@3,1
```

The Enterprise 250 OBP will use this information to create disabled status properties for each node listed in the variable `asr-disable-list`.

### “Hard” Deconfigure Override

For overriding those subsystems that require “hard” deconfiguration (CPU and memory), the OBP commands `asr-enable` and `asr-disable` are used to selectively enable or disable each subsystem.

---

**Note** – There are duplications between the soft and hard overrides. If possible, the hard override commands `asr-enable` and `asr-disable` should be used.

---

You can generate a list of valid parameters for `asr-disable` and `asr-enable` by issuing either command without parameters.

```
ok asr-disable
? Invalid subsystem name:
Known 'enable/disable' subsystem components are:
bank*          bank3          bank2          bank1          bank0
dimm15         dimm14         dimm13         dimm12         dimm11
dimm10         dimm9          dimm8          dimm7          dimm6
dimm5          dimm4          dimm3          dimm2          dimm1
dimm0          cpu*           cpu1           cpu0
ok
```

To track the status of all manual overrides, a new user command, `.asr`, is provided to summarize the current settings.

```
ok asr-disable cpu1 bank3
ok .asr
CPU0:Enabled
CPU1:Disabled
SC-MP:Enabled
Psycho@1f:Enabled
Cheerio:Enabled
SCSI:Enabled
Mem Bank0:Enabled
Mem Bank1:Enabled
Mem Bank2:Enabled
Mem Bank3:Disabled
PROM:Enabled
NVRAM:Enabled
TTY:Enabled
SuperIO:Enabled
PCI Slots:Enabled
```

## Auto-Boot Options

OpenBoot provides for an NVRAM controlled switch called `auto-boot?`, which controls whether OBP will automatically boot the operating system after each reset. The default for Sun platforms is `true`.

If a system fails power-on diagnostics, then `auto-boot?` is ignored and the system does not boot unless the user does it manually. This behavior is obviously not acceptable for a degraded boot scenario, so the Enterprise 250 OBP provides a second NVRAM-controlled switch called `auto-boot-on-error?`. This switch controls whether the system will attempt a degraded boot when a subsystem failure is detected. Both the `auto-boot?` and `auto-boot-on-error?` switches must be set to `true` to enable a degraded boot.

```
ok setenv auto-boot-on-error? true
```

---

**Note** – The default setting for `auto-boot-on-error?` is `false`. Therefore, the system will not attempt a degraded boot unless you change this setting to `true`. In addition, the system will not attempt a degraded boot in response to any fatal unrecoverable error, even if degraded booting is enabled. An example of a fatal unrecoverable error is when both of the system's CPUs have been disabled, either by failing POST or as a result of a manual user override.

---

## Reset Scenarios

The standard system reset protocol bypasses firmware diagnostics completely unless the NVRAM variable `diag-switch?` is set to `true`. The default setting for this variable is `false`.

To support ASR in Enterprise 250 servers, it is desirable to be able to run firmware diagnostics (POST/OBDiag) on any or all reset events. Rather than simply changing the default setting of `diag-switch?` to `true`, which carries with it other side effects (see the *OpenBoot 3.x Command Reference Manual*), the Enterprise 250 OBP provides a new NVRAM variable called `diag-trigger` that lets you choose which reset events, if any, will automatically engage POST/OBDiag. The `diag-trigger` variable, and its various settings are described in the following table.



---

**Note** – `diag-trigger` has no effect unless `diag-switch?` is set to `true`.

---

**TABLE 1-4** Reset Diagnostic Trigger Settings and Effects

Setting	Function
<code>power-reset</code> <i>(default)</i>	Runs diagnostics only on power-on resets.
<code>error-reset</code>	Runs diagnostics only on power-on resets, fatal hardware errors, and watchdog reset events.
<code>soft-reset</code>	Runs diagnostics on all resets (except XIR), including resets triggered by UNIX <code>init 6</code> or <code>reboot</code> commands.
<code>none</code>	Disables the automatic triggering of diagnostics by any reset event. Users can still invoke diagnostics manually by holding down the Stop and d keys when powering on the system, or by turning the front panel keyswitch to the Diagnostics position when powering on the system.

In the following example, the `diag-trigger` variable is used to trigger POST and OpenBoot diagnostics on all resets except XIR resets.

```
ok setenv diag-switch? true
ok setenv diag-trigger soft-reset
```



## Disk Drive Hot-Plug Procedures

---

The Sun Enterprise 250 server supports “hot-plugging” of internal disk drives. This hot-plug feature enables you to install a new disk drive, or remove and replace a failed disk drive, without shutting down the operating system or powering off the system. The hot-plug procedure involves software commands for preparing the system prior to removing a disk drive and for reconfiguring the operating environment after installing a new drive.



---

**Caution** – Drives should not be pulled out randomly. If the drive is active, you must stop that activity before removing the drive. This can be done without bringing down the operating system or powering down the system. The system supports hot-plugging, but there are software considerations that must be taken into account. Follow the procedures in this document when removing, replacing, and adding drives.

---

---

## Overview

Hot-plug reconfiguration or hot-plug operations cannot be performed on an active disk drive. All disk access activity must be stopped on a disk drive being removed or replaced.

In general, hot-plug reconfiguration operations involve three separate stages:

1. Preparing for hot-plug reconfiguration
2. Adding, replacing, or removing a disk drive
3. Reconfiguring the operating environment.

Three specific cases exist where the hot-plug feature is useful.

- Adding a disk drive to a system to increase storage capacity:  
See the section “Adding a Hot-Pluggable Disk Drive.
- Replacing a faulty disk drive while the system is running:  
See the section “Replacing a Faulty Hot-Pluggable Disk Drive.  
When you intend to replace a faulty drive, prepare replacement disks ahead of time to simplify replacement later.  
Each replacement disk should be formatted, labeled, and partitioned the same as the disks to be replaced, and have file systems or other application-specific preparation performed.
- Removing a drive from a system that no longer needs it:  
See the section “Removing a Hot-Pluggable Disk Drive.

---

## Adding a Hot-Pluggable Disk Drive

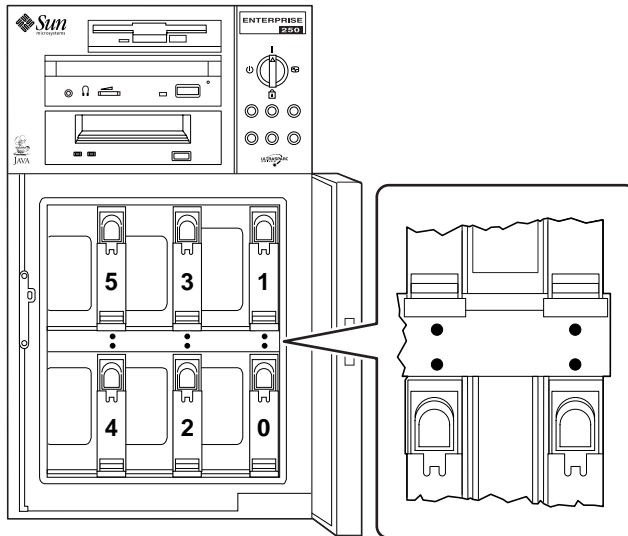
This section contains information on how to configure your system when you add a disk drive while the power is on and the operating system is running.

The way in which you add a disk drive depends on the application you are using. Each application requires that you decide where to install the new disk drive, add the drive, and then reconfigure the operating environment.

In all cases, you must select a slot, physically install the disk drive, and configure the Solaris environment to recognize the drive. Then you must configure your application to accept the new disk drive.

### **1. Select a slot for the new disk drive.**

The Enterprise 250 server’s internal disk array can accommodate up to six UltraSCSI disk drives. The figure below shows the system’s six internal disk slots. Disk slots are numbered from 0 to 5. Select any available slot for the new disk drive.



**FIGURE 2-1** Slot Numbers for Internal Disk Array

**2. Insert the new disk drive into the selected slot.**

Refer to the *Sun Enterprise 250 Server Owner's Guide* for drive installation instructions.

**3. Use the `drvconfig` command to create a new device entry for the drive in the `/devices` hierarchy:**

```
# drvconfig
```

**4. Determine the raw physical device name for the slot that you selected.**

Consult the following table.

**TABLE 2-1** Slot Physical Device Names

Disk Slot Number	Raw Physical Device Name
0	<code>/devices/pci@1f,4000/scsi@3/sd@0,0:c,raw</code>
1	<code>/devices/pci@1f,4000/scsi@3/sd@8,0:c,raw</code>
2	<code>/devices/pci@1f,4000/scsi@3/sd@9,0:c,raw</code>

**TABLE 2-1** Slot Physical Device Names

Disk Slot Number	Raw Physical Device Name
3	/devices/pci@1f,4000/scsi@3/sd@a,0:c,raw
4	/devices/pci@1f,4000/scsi@3/sd@b,0:c,raw
5	/devices/pci@1f,4000/scsi@3/sd@c,0:c,raw

**5. Use the `ssaadm insert_device` command to add the new device:**

```
# ssaadm insert_device physical_device_name
ssaadm: warning: can't quiesce
"/devices/pci@1f,4000/scsi@3/sd@b,0:c,raw": I/O error
Bus is ready for the insertion of device(s)
Insert device(s) and reconfigure bus as needed
Press RETURN when ready to continue
```

Here, *physical\_device\_name* is the full physical device name determined in Step 4.

You can safely ignore the warning message since the Enterprise 250 SCSI bus does not require quiescing.

**6. Press Return to complete the hot-plug operation.**

The `ssaadm` command creates a new device entry for the drive in the `/dev/dsk`, and `/dev/rdisk` hierarchies. The new drive is assigned a logical device name of the form `cwtxdysz`, where:

- `w` corresponds to the SCSI controller for the disk drive
- `x` corresponds to the SCSI target for the disk drive
- `y` is the logical unit number for the disk drive (always 0)
- `z` is the slice (or partition) on the disk

The logical device name assigned to the drive depends on the disk slot number where the drive is installed.

## 7. To verify that the new disk has been created, type:

```
# ls -lt /dev/dsk | more
lrwxrwxrwx 1 root root 41 Jan 30 09:07 c0t11d0s0 ->
../../../../devices/pci@1f,4000/scsi@3/sd@b,0:a
lrwxrwxrwx 1 root root 41 Jan 30 09:07 c0t11d0s1 ->
../../../../devices/pci@1f,4000/scsi@3/sd@b,0:b
lrwxrwxrwx 1 root root 41 Jan 30 09:07 c0t11d0s2 ->
../../../../devices/pci@1f,4000/scsi@3/sd@b,0:c
lrwxrwxrwx 1 root root 41 Jan 30 09:07 c0t11d0s3 ->
../../../../devices/pci@1f,4000/scsi@3/sd@b,0:d
lrwxrwxrwx 1 root root 41 Jan 30 09:07 c0t11d0s4 ->
../../../../devices/pci@1f,4000/scsi@3/sd@b,0:e
lrwxrwxrwx 1 root root 41 Jan 30 09:07 c0t11d0s5 ->
../../../../devices/pci@1f,4000/scsi@3/sd@b,0:f
lrwxrwxrwx 1 root root 41 Jan 30 09:07 c0t11d0s6 ->
../../../../devices/pci@1f,4000/scsi@3/sd@b,0:g
lrwxrwxrwx 1 root root 41 Jan 30 09:07 c0t11d0s7 ->
../../../../devices/pci@1f,4000/scsi@3/sd@b,0:h
--More--(13%)
```

The new disk and its logical device name appear at the top of the list. Check the file creation date to make sure it matches the current time and date. In the example above, the logical device name for the new disk is `c0t11d0`.

## Configuring the New Disk Drive Within Your Application

Configure the new disk drive by following the instructions for your specific application:

- “Configuring the New Disk Drive for a UNIX File System (UFS)”
- “Adding a Disk to a Solstice DiskSuite Disk Set”



---

**Caution** – These procedures should be performed only by a qualified system administrator. Performing hot-plug operations on an active disk drive may result in data loss if performed incorrectly.

---

# Configuring the New Disk Drive for a UNIX File System (UFS)

Use the following procedure to configure a slice (single physical partition) on a disk to be used with a UFS file system. For instructions about adding a file system to a Solstice™ DiskSuite™ (SDS) logical disk, refer to the documentation that came with your application.

## 1. Verify that the device label meets your requirements.

You can use the `prtvtoc` command to inspect the label for your disk. To modify the label, use the `format` command. Refer to the `prtvtoc(1M)` and `format(1M)` man pages for more information.

## 2. Select a disk slice for your UFS file system and create a new file system on the slice:

```
# newfs /dev/rdisk/cwtxdysz
```

For example: `newfs /dev/rdisk/c0t11d0s2`

Refer to the `newfs(1M)` man page for more information.

## 3. If necessary, create a mount point for the new file system:

```
# mkdir mount_point
```

where `mount_point` is a fully qualified path name. Refer to the `mount(1M)` man page for more information.

## 4. After the file system and mount point have been created, modify the `/etc/vfstab` file to reflect the new file system.

See the `vfstab(4)` man page for more details.

## 5. Mount the new file system using the `mount` command:

```
# mount mount_point
```

where `mount_point` is the directory you created.

The file system is ready to be used.



## Adding a Disk to a Solstice DiskSuite Disk Set

You can use any disk you add to the system for Solstice DiskSuite (SDS) new or existing metadevices.

Refer to the Solstice DiskSuite documentation for information on configuring the disk drive.

---

## Replacing a Faulty Hot-Pluggable Disk Drive

This section contains information on configuring your system to replace a disk drive while the power is on and the operating system is running.

The way in which you replace a faulty disk drive depends on the application you are using. Each application is different, but requires that you:

1. Determine which disk drive is failing or has failed
2. Remove the disk
3. Add the replacement drive
4. Reconfigure the operating environment.

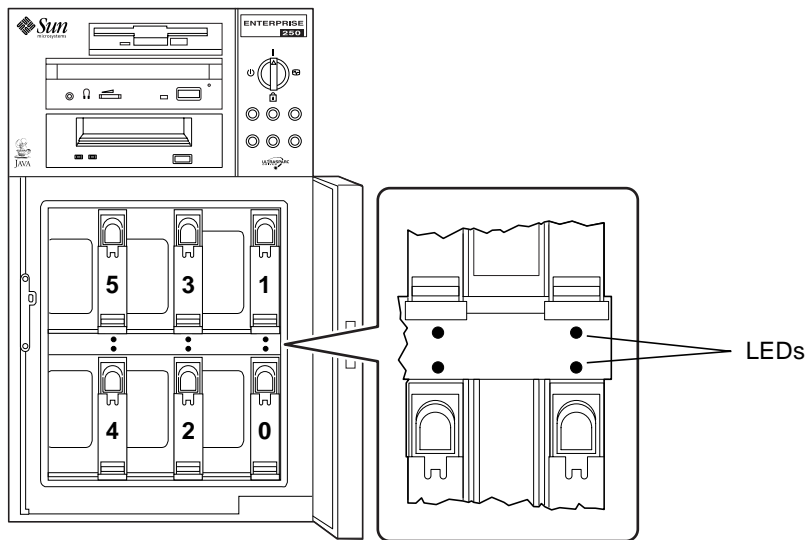
In all cases you must stop any activity or applications on the disk; unmount it; physically remove the old drive and install the new one; and configure the Solaris environment to recognize the drive. Then you must configure your application to accept the new disk drive.

## Prepare Spare Drives

If possible, prepare replacement disk drives in advance. Each replacement disk drive should be formatted, labeled, and partitioned the same as the disk it will replace. See the documentation for your application for instructions on how to format and partition the disk, and add that disk to your application.

# Identifying the Faulty Disk Drive

Disk errors may be reported in a number of different ways. Often you can find messages about failing or failed disks in your system console. This information is also logged in the `/usr/adm/messages` file(s). These error messages typically refer to a failed disk drive by its physical device name (such as `/devices/pci@1f,4000/scsi@3/sd@b,0`) and its UNIX device instance name (such as `sd11`). In some cases, a faulty disk may be identified by its logical device name (such as `c0t11d0`). In addition, some applications may report a disk slot number (0 through 5) or activate an LED located next to the disk drive itself (see following figure).



**FIGURE 2-2** Disk Slot Numbers and LED Locations

In order to perform a disk hot-plug procedure, you need to know the slot number of the faulty disk (0 through 5) and its logical device name (for example, `c0t11d0`). If you know the disk slot number, it is possible to determine the logical device name, and vice versa. It is also possible to determine both the disk slot number and the logical device name from a physical device name (such as `/devices/pci@1f,4000/scsi@3/sd@b,0`).

To make the necessary translation from one form of disk identifier to another, see the Chapter *Mapping Between Logical and Physical Device Names*. Once you have determined both the disk slot number and logical device name, you are ready to continue with this procedure.

# Replacing the Disk Drive Within Your Application

Continue the disk replacement by following the instructions for your specific application.

- “UNIX File System (UFS)
- “Solstice DiskSuite

## UNIX File System (UFS)

The following procedure describes how to deconfigure a disk being used by one or more UFS file systems.



---

**Caution** – These procedures should be performed only by a qualified system administrator. Performing hot-plug operations on an active disk drive can result in data loss if performed incorrectly.

---

**1. Type `su` and your superuser password.**

**2. Identify activities or applications attached to the device you plan to remove.**

Commands to use are `mount`, `showmount -a`, and `ps -ef`. See the `mount(1M)`, `showmount(1M)`, and `ps(1)` man pages for more details.

For example, where the controller number is 0 and the target ID is 11:

```
# mount | grep c0t11
/export/home1 on /dev/dsk/c0t11d0s2 setuid/read/write on
# showmount -a | grep /export/home1
cinnamon:/export/home1/archive
austin:/export/home1
swlab1:/export/home1/doc
# ps -f | grep c0t11
root 1225 450 4 13:09:58 pts/2 0:00 grep c0t11
```

In this example, the file system `/export/home1` on the faulty disk is being remotely mounted by three different systems—`cinnamon`, `austin`, and `swlab1`. The only process running is `grep`, which has finished.

**3. Stop any activity or application processes on the file systems to be deconfigured.**

**4. Back up your system.**

## 5. Determine what file system(s) are on the disk:

```
# mount | grep cwtX
```

For example, if the device to be removed is `c0t11d0`, enter the following:

```
# mount | grep c0t11
/export/home (/dev/dsk/c0t11d0s7 ): 98892 blocks 142713 files
/export/home1 (/dev/dsk/c0t11d0s5 ): 153424 blocks 112107 files
```

## 6. Determine and save the partition table for the disk.

If the replacement disk is the same type as the faulty disk, you can use the `format` command to save the partition table of the disk. Use the `save` command in `format` to save a copy of the partition table to the `/etc/format.dat` file. This will allow you to configure the replacement disk so that its layout matches the current disk.

Refer to the `format(1M)` man page for more information.

## 7. Unmount any file systems on the disk.

For each file system returned, type:

```
# umount file_system
```

where `file_system` is the first field for each line returned in Step 5.

For example:

```
# umount /export/home
# umount /export/home1
```

---

**Note** – If the file system(s) are on a disk that is failing or has failed, the `umount` operation may not complete. A large number of error messages may be displayed in the system console and in the `/var` directory during the `umount` operation. If the `umount` operation does not complete, you may have to restart the system.

---

**8. Use the `ssaadm replace_device` command to take the device offline:**

```
# ssaadm replace_device logical_device_name
ssaadm: warning: can't quiesce "/dev/rdisk/c0t11d0s2": I/O error
Bus is ready for the replacement of device
Replace device and reconfigure bus as needed
Press RETURN when ready to continue
```

Here, *logical\_device\_name* is the full logical device name of the drive to be removed (`/dev/rdisk/c0t11d0s2`). You must specify slice 2, which represents the entire disk. Note that this command also accepts a physical device name as an alternative. You can safely ignore the warning message since the Enterprise 250 SCSI bus does not require quiescing.

**9. Remove the faulty disk drive and insert the replacement drive in its place.**

Refer to the *Sun Enterprise 250 Server Owner's Guide* for drive removal and replacement instructions.

**10. Press Return to complete the hot-plug operation.**

The `ssaadm` command brings the replacement drive back online.

**11. Verify that the device's partition table satisfies the requirements of the file system(s) you intend to re-create.**

You can use the `prtvtoc` command to inspect the label for your device. If you need to modify the label, use the `format` command. Refer to the `prtvtoc(1M)` and `format(1M)` man pages for more information.

For example:

```
# prtvtoc /dev/rdisk/cwtxdysz
```

If you have saved a disk partition table using the `format` utility and the replacement disk type matches the old disk type, then you can use the `format` utility's `partition` section to configure the partition table of the replacement disk. See the `select` and `label` commands in the `partition` section.

If the replacement disk is of a different type than the disk it replaced, you can use the partition size information from the previous disk to set the partition table for the replacement disk. Refer to the `prtvtoc(1M)` and `format(1M)` man pages for more information.

You have defined your disk's partition table and have identified the disk slice on which to build your UFS file system.

12. Once you have selected a disk slice for your UFS file system, check and/or create a file system on the slice:

```
# fsck /dev/rdisk/cwtXdysz
# newfs /dev/rdisk/cwtXdysz
```

13. Mount the new file system using the `mount` command:

```
# mount mount_point
```

where *mount\_point* is the directory on which the faulty disk was mounted.

The new disk is ready to be used. You can now restore data from your backups.

## Solstice DiskSuite

The following procedure describes how to replace a disk in use by Solstice DiskSuite. Refer to the Solstice DiskSuite documentation for more information.



---

**Caution** – These procedures should be performed only by a qualified system administrator. Performing hot-plug operations on an active disk drive can result in data loss if performed incorrectly.

---

1. **Back up your system.**
2. **Type `su` and your superuser password.**
3. **If possible, save the partition table for the disk you intend to replace.**  
If the disk label can still be read, save the disk partitioning at this time.

---

**Note** – Save all the disk partitioning information immediately after configuring metadevices or file systems for use when recovering from device failure later.

---

Use the `prtvtoc` command to save the partition information.

```
# prtvtoc /dev/rdisk/cwtxdys0 > saved_partition_table_file
```

For example:

```
# prtvtoc /dev/rdisk/c0t11d0s0 > /etc/c0t11d0s0.vtoc
```

#### 4. Identify metadevices or applications using the device you plan to remove.

For example:

```
# metadb | grep c0t11d0
# metastat | grep c0t11d0
# mount | grep c0t11d0
```

Save the output of the commands to reconstruct the metadevice configuration after you replace the disk.

#### 5. Delete database replicas.

If there are database replicas on the disk, these must be deleted. First record the size and number of database replicas on each slice; then delete them.

```
# metadb -d cwtxdysz
```

For example:

```
# metadb -d c0t11d0s0
```

#### 6. Detach submirrors.

If any slices of the disk are used by submirrors, the submirrors should be detached. For example:

```
# metadetach d20 d21
```

## 7. Delete hotspares.

If any slices are used by hotspare pools, remove them. Record the hotspare pools containing the slices; then delete them. For example:

```
# metahs -d all c0t11d0s1
```

## 8. Terminate all other metadvice activity on the disk.

Check `metastat` output for other slices of the disk used by metadevices that cannot be detached (stripes not in mirrors, etc.). These metadevices must be unmounted if they contain file systems, or they must otherwise be taken off line.

Refer to the `prtvtoc(1M)` man page for more information.

## 9. Unmount any file systems on the disk.

---

**Note** – If the file system(s) are on a disk that is failing or has failed, the `umount` operation may not complete. A large number of error messages may be displayed in the system console and in the `/var` directory during the `umount` operation. If the `umount` operation does not complete, you may have to restart the system.

---

For each file system returned, type:

```
# umount file_system
```

where *file\_system* is the first field for each line returned in Step 4.

For example:

```
# umount /export/home
# umount /export/home1
```



**10. Use the `ssaadm replace_device` command to take the device offline:**

```
# ssaadm replace_device logical_device_name
ssaadm: warning: can't quiesce "/dev/rdisk/c0t11d0s2": I/O error
Bus is ready for the replacement of device
Replace device and reconfigure bus as needed
Press RETURN when ready to continue
```

Here, *logical\_device\_name* is the full logical device name of the drive to be removed (`/dev/rdsk/c0t11d0s2`). You must specify slice 2, which represents the entire disk. Note that this command also accepts a physical device name as an alternative.

You can safely ignore the warning message since the Enterprise 250 SCSI bus does not require quiescing.

**11. Remove the faulty disk drive and insert the replacement drive in its place.**

Refer to the *Sun Enterprise 250 Server Owner's Guide* for drive removal and replacement instructions.

**12. Press Return to complete the hot-plug operation.**

The `ssaadm` command brings the replacement drive back online.

**13. Restore the disk partitioning.**

If you have saved the disk partitioning to a file, you may restore it with `fmthard`. For example:

```
# fmthard -s /etc/c0t11d0s0.vtoc /dev/rdsk/c0t11d0s0
```

If you have not saved the disk partitioning, use the `format (1M)` or `fmthard(1M)` command to repartition the disk.

**14. Replace any database replicas.**

For example:

```
# metadb -a -l 2000 -c 2 c0t11d0s0
```

**15. Reattach any submirrors.**

For example:

```
# metattach d20 d21
```

**16. Re-create hot spares for each hot spare pool that contained a slice on the new disk.**

For example:

```
# metahs -a hsp001 c0t11d0s1
```

**17. Fix any broken metadevices, using slices from the new disk.**

If the disk to be replaced had caused any metadevices to go into the maintenance state, these metadevices can be repaired by re-enabling the slices.

```
# metareplace -e mirror_or_RAID5_metadevice cwtxdysz
```

**18. Remount any file systems and restart any applications that were using metadevices that could not be taken off line.**

```
# mount file_system
```

Refer to the Solstice DiskSuite documentation for more information.

---

## Removing a Hot-Pluggable Disk Drive

This section contains information on how to configure your system to remove a disk drive while the power is on and the operating system is running. Use the procedures in this chapter if you do *not* intend to replace the disk drive.

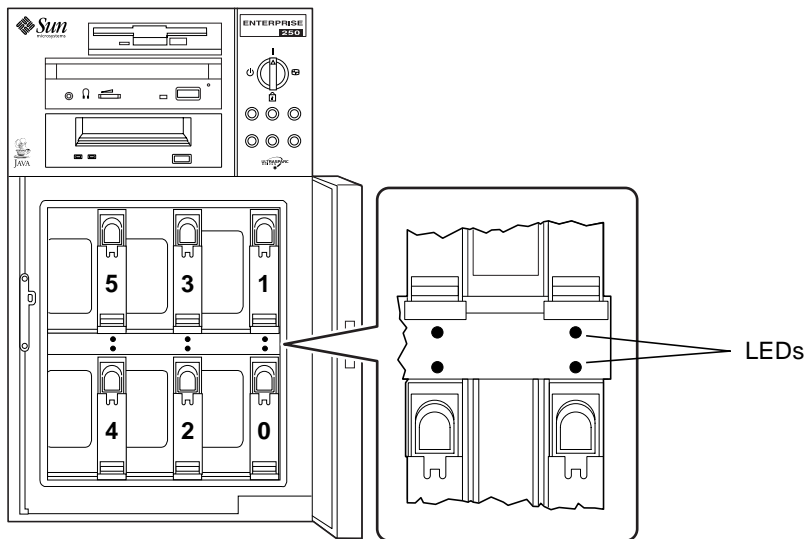
The way in which you remove a disk drive depends on the application you are using. Each application is different, but requires that you:

1. Select the disk drive
2. Remove the disk
3. Reconfigure the operating environment.

In all cases you must select the disk and stop any activity or applications on it, unmount it, physically remove the drive, and configure the Solaris environment to recognize that the drive is no longer there. Then you must configure your application to operate without this device in place.

# Identifying the Faulty Disk Drive

Disk errors may be reported in a number of different ways. Often you can find messages about failing or failed disks in your system console. This information is also logged in the `/usr/adm/messages` file(s). These error messages typically refer to a failed disk drive by its UNIX physical device name (such as `/devices/pci@1f,4000/scsi@3/sd@b,0`) and its UNIX device instance name (such as `sd11`). In some cases, a faulty disk may be identified by its UNIX logical device name, such as `c0t11d0`. In addition, some applications may report a disk slot number (0 through 5) or activate an LED located next to the disk drive itself (see the following figure).



**FIGURE 2-3** Disk Slot Numbers and LED Locations

In order to perform a disk hot-plug procedure, you need to know the slot number of the faulty disk (0 through 5) and its logical device name (for example, `c0t11d0`). If you know the disk slot number, it is possible to determine the logical device name, and vice versa. It is also possible to determine both the disk slot number and the xlogical device name from a physical device name (such as `/devices/pci@1f,4000/scsi@3/sd@b,0`).

To make the necessary translation from one form of disk identifier to another, see the chapter, “Mapping From UNIX Logical Name to Disk Slot Number. Once you have determined both the disk slot number and logical device name, you are ready to continue with this procedure.

# Removing a Disk Drive From Your Application

Continue the hot disk removal by following the instructions for your specific application:

- “UNIX File System (UFS)
- “Solstice DiskSuite

## UNIX File System (UFS)

The following procedure describes how to remove a disk being used by one or more UFS file systems.

1. **Type `su` and your superuser password.**
2. **Identify activities or applications attached to the device you plan to remove.**

Commands to use are `mount`, `showmount -a`, and `ps -ef`. See the `mount(1M)`, `showmount(1M)`, and `ps(1)` man pages for more details.

For example, where the controller number is 0 and the target ID is 11:

```
# mount | grep c0t11
/export/home1 on /dev/dsk/c0t11d0s2 setuid/read/write on
# showmount -a | grep /export/home1
cinnamon:/export/home1/archive
austin:/export/home1
swlab1:/export/home1/doc
# ps -f | grep c0t11
root 1225 450 4 13:09:58 pts/2 0:00 grep c0t11
```

In this example, the file system `/export/home1` on the faulty disk is being remotely mounted by three different systems—`cinnamon`, `austin`, and `swlab1`. The only process running is `grep`, which has finished.

3. **Stop any activity or application processes on the file systems to be deconfigured.**
4. **Back up your system.**
5. **Determine what file system(s) are on the disk:**

```
# mount | grep cwtx
```

6. **Unmount any file systems on the disk.**

---

**Note** – If the file system(s) are on a disk that is failing or has failed, the `umount` operation may not complete. A large number of error messages may be displayed in the system console and in the `/var` directory during the `umount` operation. If the `umount` operation does not complete, you may have to restart the system.

---

For each file system returned, type:

```
# umount file_system
```

where `file_system` is the first field for each line returned in Step 5.

For example:

```
# umount /export/home
# umount /export/home1
```

**7. Use the `ssaadm remove_device` command to take the device offline:**

```
# ssaadm remove_device logical_device_name
ssaadm: warning: can't quiesce "/dev/rdisk/c0t11d0s2": I/O error
Bus is ready for the removal of device
Remove device and reconfigure bus as needed
Press RETURN when ready to continue
```

Here, `logical_device_name` is the full logical device name for the drive to be removed (`/dev/rdisk/c0t11d0s2`, for example). You must specify slice 2, which represents the entire disk. Note that this command also accepts a physical device name as an alternative.

You can safely ignore the warning message since the Enterprise 250 SCSI bus does not require quiescing.

**8. Remove the disk drive from its slot.**

Refer to the *Sun Enterprise 250 Server Owner's Guide* for drive removal instructions.

**9. Press Return to complete the hot-plug operation.**

The `ssaadm` command deletes the symbolic links for the device in the `/dev/dsk`, and `/dev/rdsk` hierarchies.

# Solstice DiskSuite

The following procedure describes how to deconfigure a disk in use by Solstice DiskSuite software. For more information, refer to the Solstice DiskSuite documentation.

1. **Back up your system.**
2. **Type `su` and your superuser password.**
3. **Identify metadevices or applications using the device you plan to remove.**

For example:

```
# metadb | grep c0t11d0
# metastat | grep c0t11d0
# mount | grep c0t11d0
```

4. **Delete database replicas.**

If there are database replicas on the disk, these must be deleted. For example:

```
# metadb -d c0t11d0s0
```

5. **Replace slices or clear metadevices.**

If any slices of the disk are in use by submirrors or within RAID metadevices, they can be replaced by other available slices. For example:

```
# metareplace d20 c0t11d0s1 c0t8d0s1
```

If there are no replacement slices available, the metadevices must be cleared. For example:

```
# metaclear d21
```

6. **Replace slices or clear hotspares.**

If any slices of the disk are used by hotspare pools, they can be replaced by other available slices. For example:

```
# metahs -r all c0t11d0s1 c0t8d0s1
```

7. **Unmount any file systems on the disk.**

---

**Note** – If the file system(s) are on a disk that is failing or has failed, the `umount` operation may not complete. A large number of error messages may be displayed in the system console and in the `/var` directory during the `umount` operation. If the `umount` operation does not complete, you may have to restart the system.

---

For each file system, type:

```
# umount file_system
```

For example:

```
# umount /export/home
# umount /export/home1
```

Refer to the Solstice DiskSuite documentation for more information.

**8. Use the `ssaadm remove_device` command to take the device offline:**

```
# ssaadm remove_device logical_device_name
ssaadm: warning: can't quiesce "/dev/rdisk/c0t11d0s2": I/O error
Bus is ready for the removal of device
Remove device and reconfigure bus as needed
Press RETURN when ready to continue
```

Here, *logical\_device\_name* is the full logical device name for the drive to be removed (`/dev/rdisk/c0t11d0s2`, for example). You must specify slice 2, which represents the entire disk. Note that this command also accepts a physical device name as an alternative.

You can safely ignore the warning message since the Enterprise 250 SCSI bus does not require quiescing.

**9. Remove the disk drive from its slot.**

Refer to the *Sun Enterprise 250 Server Owner's Guide* for drive removal instructions.

**10. Press Return to complete the hot-plug operation.**

The `ssaadm` command deletes the symbolic links for the device in the `/dev/dsk`, and `/dev/rdsk` hierarchies.





# Mapping Between Logical and Physical Device Names

---

This chapter contains procedures for translating between the various logical and physical names used to identify internal disk drives in a Sun Enterprise 250 server running the Solaris 2.x operating environment.

---

## Overview

The Enterprise 250 server can accommodate up to six internal UltraSCSI disk drives. Each drive is labeled with a disk slot number between 0 and 5.

In order to perform a disk hot-plug procedure, you must know the slot number of the faulty disk (0 through 5) and its logical device name (for example, `c0t11d0`). If you know the disk slot number, it is possible to determine the logical device name, and vice versa. It is also possible to determine both the disk slot number and the logical device name if you know the physical device name (such as `/devices/pci@1f,4000/scsi@3/sd@b,0`). Physical device names are typically provided in SCSI error messages generated by software.

This chapter provides procedures for:

- “Mapping From Error Message to Disk Slot Number and UNIX Logical Name
- “Mapping From UNIX Logical Name to Disk Slot Number
- “Mapping From Disk Slot Number to UNIX Logical Name

---

**Note** – If you use Solstice™ SyMON™ software to monitor your Enterprise 250 server, there is no need to perform the following procedures. Solstice SyMON displays disk slot numbers together with physical and logical device names in its Physical View and Logical View consoles. For additional details, please see the *Solstice SyMON User's Guide* supplied with your version of Solstice SyMON.

---

## Mapping From Error Message to Disk Slot Number and UNIX Logical Name

This section describes how to translate a UNIX physical device name provided in a SCSI error message to a UNIX logical device name and a disk slot number.

### 1. Determine the UNIX physical device name from the SCSI error message.

SCSI error messages are typically displayed in the system console and logged in the `/usr/adm/messages` file.

```
WARNING: /pci@1f,4000/scsi@3/sd@b,0 (sd11)
Error for Command: read(10)      Error level: Retryable
Requested Block: 3991014        Error Block: 3991269
Vendor: FUJITSU                 Serial Number: 9606005441
Sense Key: Media Error
ASC: 0x11 (unrecovered read error), ASCQ: 0x0, FRU: 0x0
```

In the example SCSI error message above, the UNIX physical device name is `/pci@1f,4000/scsi@3/sd@b`.

**2. Determine the UNIX logical device name by listing the contents of the /dev/rdisk directory.**

Use the `grep` command to filter the output for any occurrence of the UNIX physical device name determined in Step 1:

```
% ls -l /dev/rdisk | grep /pci@1f,4000/scsi@3/sd@b
lrwxrwxrwx  1 root  root          45 Nov 24 10:41 c0t11d0s0 -
> ../../devices/pci@1f,4000/scsi@3/sd@b,0:a,raw
lrwxrwxrwx  1 root  root          45 Nov 24 10:41 c0t11d0s1 -
> ../../devices/pci@1f,4000/scsi@3/sd@b,0:b,raw
lrwxrwxrwx  1 root  root          45 Nov 24 10:41 c0t11d0s2 -
> ../../devices/pci@1f,4000/scsi@3/sd@b,0:c,raw
lrwxrwxrwx  1 root  root          45 Nov 24 10:41 c0t11d0s3 -
> ../../devices/pci@1f,4000/scsi@3/sd@b,0:d,raw
lrwxrwxrwx  1 root  root          45 Nov 24 10:41 c0t11d0s4 -
> ../../devices/pci@1f,4000/scsi@3/sd@b,0:e,raw
lrwxrwxrwx  1 root  root          45 Nov 24 10:41 c0t11d0s5 -
> ../../devices/pci@1f,4000/scsi@3/sd@b,0:f,raw
lrwxrwxrwx  1 root  root          45 Nov 24 10:41 c0t11d0s6 -
> ../../devices/pci@1f,4000/scsi@3/sd@b,0:g,raw
lrwxrwxrwx  1 root  root          45 Nov 24 10:41 c0t11d0s7 -
> ../../devices/pci@1f,4000/scsi@3/sd@b,0:h,raw
```

The resulting output indicates the associated UNIX logical device name. In this example, the logical device name is `c0t11d0`.

**3. Determine the disk slot number using the `prtconf` command.**

Substitute the string `disk@` for `sd@` in the physical device name determined in Step 1. The result in this example is `/pci@1f,4000/scsi@3/disk@b`.

Use the `grep` command to find this name in the output of the `prtconf` command:

```
% prtconf -vp | grep /pci@1f,4000/scsi@3/disk@b
disk4:  '/pci@1f,4000/scsi@3/disk@b,0'
disk-slot4:  '/pci@1f,4000/scsi@3/disk@b'
```

The resulting output indicates the corresponding disk slot number. In this example, the disk slot number is 4.

If the output does not provide a slot number, the device is either a removable media device (CD-ROM or tape drive) or an external device.

---

# Mapping From UNIX Logical Name to Disk Slot Number

This section describes how to translate from a known UNIX logical device name such as `c0t11d0s0` to a disk slot number (0 through 5).

This example assumes a known UNIX logical device name of `c0t11d0s0`.

## 1. Determine the UNIX physical device name from the UNIX logical device name.

Use the `ls -l` command to display the link for the logical device name in the `/dev/dsk` directory:

```
% ls -l /dev/dsk/c0t11d0s0
lrwxrwxrwx  1 root      root          41 Nov 24 10:41
/dev/dsk/c0t11d0s0 -> ../../devices/pci@1f,4000/scsi@3/sd@b,0:a
```

The resulting output shows the UNIX physical device name associated with the logical device name. In this case, the corresponding physical device name is `/pci@1f,4000/scsi@3/sd@b`.

## 2. Determine the disk slot number using the `prtconf` command.

Substitute the string `disk@` for `sd@` in the physical device name determined in Step 1. The result in this example is `/pci@1f,4000/scsi@3/disk@b`.

Use the `grep` command to find this name in the output of the `prtconf` command:

```
% prtconf -vp | grep /pci@1f,4000/scsi@3/disk@b
disk4:  '/pci@1f,4000/scsi@3/disk@b,0'
disk-slot4:  '/pci@1f,4000/scsi@3/disk@b'
```

The resulting output indicates the corresponding disk slot number. In this example, the disk slot number is 4.

If the output does not provide a slot number, the device is either a removable media device (CD-ROM or tape drive) or an external device.

---

# Mapping From Disk Slot Number to UNIX Logical Name

This section describes how to translate from a known disk slot number (0 through 5) to a UNIX logical device name such as `c0t11d0`.

The example in this procedure assumes a known disk slot number of 4.

## 1. Determine the UNIX physical device name using the `prtconf` command.

Use the `grep` command to filter the `prtconf` output for any occurrence of the disk slot number:

```
% prtconf -vp | grep disk-slot4
    disk-slot4:  '/pci@1f,4000/scsi@3/disk@b'
```

In this example, the physical name associated with disk slot number 4 is `/pci@1f,4000/scsi@3/disk@b`. To translate this to a UNIX physical device name, substitute `sd@` for `disk@`. The resulting UNIX physical device name is `/pci@1f,4000/scsi@3/sd@b`.

**2. Determine the UNIX logical device name by listing the contents of the /dev/rdisk directory.**

Use the `grep` command to filter the output for any occurrence of the UNIX physical device name determined in Step 1:

```
% ls -l /dev/rdisk | grep /pci@1f,4000/scsi@3/sd@b
lrwxrwxrwx  1 root    root          45 Nov 24 10:41 c0t11d0s0 -
> ../../devices/pci@1f,4000/scsi@3/sd@b,0:a,raw
lrwxrwxrwx  1 root    root          45 Nov 24 10:41 c0t11d0s1 -
> ../../devices/pci@1f,4000/scsi@3/sd@b,0:b,raw
lrwxrwxrwx  1 root    root          45 Nov 24 10:41 c0t11d0s2 -
> ../../devices/pci@1f,4000/scsi@3/sd@b,0:c,raw
lrwxrwxrwx  1 root    root          45 Nov 24 10:41 c0t11d0s3 -
> ../../devices/pci@1f,4000/scsi@3/sd@b,0:d,raw
lrwxrwxrwx  1 root    root          45 Nov 24 10:41 c0t11d0s4 -
> ../../devices/pci@1f,4000/scsi@3/sd@b,0:e,raw
lrwxrwxrwx  1 root    root          45 Nov 24 10:41 c0t11d0s5 -
> ../../devices/pci@1f,4000/scsi@3/sd@b,0:f,raw
lrwxrwxrwx  1 root    root          45 Nov 24 10:41 c0t11d0s6 -
> ../../devices/pci@1f,4000/scsi@3/sd@b,0:g,raw
lrwxrwxrwx  1 root    root          45 Nov 24 10:41 c0t11d0s7 -
> ../../devices/pci@1f,4000/scsi@3/sd@b,0:h,raw
```

The resulting output indicates the associated UNIX logical device name. In this example, the logical device name is `c0t11d0`.